Statistisches Modellieren – Zusammenfassung

Einführung – «Daten verlangen keine Gerechtigkeit!» – S. 9, Skript STMO Prof. Dr. Andreas Ruckstuhl

Statistische Modellierung heisst **mittels Daten** ein **Modell** für die **Realität erstellen**. Das Modell soll die **Realität beschreiben**, **vorhersagen** oder kausal **analysieren**. Mit einem statistischen Modell lassen sich **Genauigkeitsangaben** machen.

Notationen – dat <- read.table("C:/Users/denis/.../Daten.dat", header = T)

- ⟨ ⟩ = Funktionsargumente
- { } = Mengen
- ( ) = Priorität der Rechenoptionen
- [ ] = Vektoren/**Matrizen**

Regressionsanalyse – «All models are wrong, but some are useful» – George E. P. Box

- Ist eine statistische Methode, um die **Beziehung** zwischen (Mess-) **Grössen** zu **untersuchen** und zu **modellieren**.
- Modellvorstellung aufbauen: Modell mit unbekannten Parametern.

Beispiel Sprengungen beim Bau eines Strassentunnel

Erschütterung der Häuser darf einen bestimmten Wert nicht überschreiten. Erschütterung hängt ab von Ladung, Distanz vom Sprengort, Untergrundmaterial, Ort der Sprengung, etc. Die genaue Funktion zur Berechnung der Grösse der Erschütterung ist unbekannt. Schätzung der Erschütterung aus Daten.

- **Zielgrösse y (response variable)** → die Erschütterung
- Hängt über Funktion h von den **erklärenden Variablen** $x^{(1)}, \dots, x^{(m)}$ **(predictor variables)** ab → Ladung, Distanz, etc.
- Im Idealfall sollte für jede **Beobachtung/Versuch** i gelten $y_i = h\langle x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(m)}\rangle$. Diese Formel existiert leider nicht!
- Formel gilt ungefähr, (Abweichungen zufällig) deshalb: $\boldsymbol{Y_i = h\langle x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(m)}\rangle + E_i}$ und nennt die E_i **Zufallsfehler**.
- Vorstellungen über Grösse solcher Abweichungen werden mit einer Wahrscheinlichkeits-Verteilung formuliert.
- Funktion $\boldsymbol{h\langle\,\rangle}$ beschreibt oft eine Gerade; mit lin. Regression lassen sich auch nicht lineare Kurven an Daten anpassen!

Einfache lineare Regression

- Besteht nur aus einer erklärenden Variable $x_i^{(1)}$. Gerade ist einfachste Funktion, die eine Abhängigkeit ausdrücken kann: $\boldsymbol{y_i = \alpha + \beta x_i}$, wobei $\boldsymbol{\alpha}$ = **Achsenabschnitt** (Schnittpunkt mit der y-Achse) und $\boldsymbol{\beta}$ = **Steigung**
- modellieren die Abweichungen von der Geraden mit einer Zufallsvariablen: $\boldsymbol{Y_i = \alpha + \beta x_i + E_i}$
- Wir nehmen an, dass die Abweichungen E_i
  - alle eine bestimmte gleiche Verteilung haben &
  - stochastisch unabhängig (insbesondere unkorreliert) sind.
- I.d.R. für Abweichung E_i **Annahme Normalverteilung** mit $\boldsymbol{E(X) = 0}$ und $\boldsymbol{Var(X) = \sigma^2}$ → $\boldsymbol{E_i \sim \mathcal{N}\langle 0,\ \sigma^2\rangle}$
- Modell erst dann konkret, wenn die Parameter α, β und σ festgelegt sind.
- Daraus folgt, dass auch die Zielvariable Y_i eine Zufallsvariable ist, weshalb ein grosser Buchstabe verwendet wird.
- Zielvariable Y_i ist Normalverteilt mit konstanter Varianz σ^2, jedoch ist Erwartungswert für jede Beobachtung anders (d.h. $= \alpha + \beta x_i$). Das beobachte y_i entsteht dann aus einer Realisation e_i der Zufallsvariable E_i, also $\boldsymbol{y_i = \alpha + \beta x_i + e_i}$

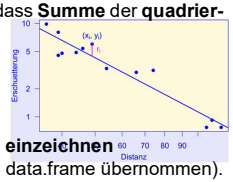
Schätzung der Parameter

- Die Funktionen, die den Daten die best-passenden Werte zuordnen, heissen Schätzfunktionen oder Schätzungen. Herleitung Schätzung nach Prinzip **Kleinsten Quadrate (KQ)**: Parameter α, β so bestimmt, dass **Summe** der **quadrierten Abweichungen**: $\sum_{i=1}^n r_i^2$ mit $\boldsymbol{r_i = y_i - (\alpha + \beta x_i)}$ **minimal** wird. $\boldsymbol{\hat\beta = \frac{\sum_{i=1}^n (Y_i - \bar Y)(x_i - \bar x)}{\sum_{i=1}^n (x_i - \bar x)^2}}$
- Die Schätzfunktionen lauten dann $\boldsymbol{\hat\alpha = \bar Y - \hat\beta\,\bar x}$ und
  Spr.lm <- lm(lErsch ~ lDist, data = SprengS1) Übergabe **data.frame** an Bf lm()
  coef(Spr.lm) (**Zielgrösse (Y) ~ erkl. Variable (X)**)

(Intercept)	*lDist*
8.9791	*-1.9235*

Geschätzte Grade lautet: $lErsch = 8.9791 + (-1.9235) \cdot lDist$

plot(lErsch ~ lDist, data = SprengS1); abline(Spr.lm) #Linie einzeichnen



- Intercept = Achsenabschnitt, lDist = Steigung der Geraden (lDist wird von Variabelname des data.frame übernommen).
- **Weiterer Parameter** im Modell, die Varianz $\boldsymbol{\sigma^2 = var\langle E_i\rangle}$ der **zufälligen Abweichungen**. Die zufälligen Fehler $\boldsymbol{E_i}$ können **weder** direkt beobachtet noch aus $E_i = Y_i - (\alpha + \beta x_i)$ hergeleitet werden, da α und β unbekannt sind. Bekannt sind «Näherungswerte» für E_i (**Residuen**) $\boldsymbol{R_i := Y_i - (\hat\alpha + \hat\beta x_i)}$, erwartungstreue Schätzung für σ^2 ist: $\boldsymbol{\hat\sigma^2 = \frac{1}{n-2}\sum_{i=1}^n R_i^2}$
- Die Kleinste-Quadrate-Schätzungen $\hat\alpha$ & $\hat\beta$ sind **erwartungstreu** und **normalverteilt** mit Varianzen $\boldsymbol{\sigma^{(\beta)2} = \frac{\sigma^2}{SS_X}}$,
  $\boldsymbol{\sigma^{(\alpha)2} = \sigma^2\left(\frac{1}{n} + \frac{\bar x^2}{SS_X}\right)}$ wobei **Quadratsumme** $\boldsymbol{SS_X = \sum_{i=1}^n (x_i - \bar x)^2}$. Sie ist die **effizientesten** Schätzungen, **sofern** die **Zufallsfehler unabhängig** sind und alle die gleiche **Normalverteilung** $\boldsymbol{\mathcal{N}\langle 0,\ \sigma^2\rangle}$ haben. Treffen diese Voraussetzungen **nicht** zu, kann Methode unzuverlässig werden, womit andere Schätzverfahren besser geeignet sind.

Statistisches Testverfahren (Vorgehen gemäss Modul GSTAT)

- Sind Daten mit Modell mit (teilweise) vorgegebenen Parametern verträglich?
- Unter H_0: T hat eine t-Verteilung mit $\boldsymbol{n - 2}$ Freiheitsgraden und der Teststatistik: $\boldsymbol{T = \frac{\hat\beta - \beta_0}{se\langle\hat\beta\rangle}}$, mit $\boldsymbol{se\langle\hat\beta\rangle = \sqrt{\frac{\hat\sigma^2}{SS_X}}}$

Hypothesentest Beispiel Kirschbäume: $lV = \log(Volumen) = y$, $lD = \log(Durchmesser) = x_1$, $lH = \log(Höhe) = x_2$

Modell: $lV = \beta_0 + \beta_1 lD + \beta_2 lH + E$ → Modellidee mit Kegelvolumen Grundlage: $V = \pi \cdot R^2 \cdot \frac{h}{3} = \pi \cdot \left(\frac{D}{2}\right)^2 \cdot \frac{H}{3} = \frac{\pi}{(2^2 \cdot 3)} \cdot D^2 \cdot H$

Nun Modellidee in Modellform von oben bringen: $\log(V) = \log\left(\frac{\pi}{12}\right) + 2 \cdot \log(D) + \log(H) = -1.340177 + 2 \cdot \log(D) + \log(H)$

Testen, ob Modellidee $\beta_1 = 2$ und $\beta_2 = 1$ sein kann → Nullhypothese: $lV = \beta_0 + 2lD + 1lH + E$; schätzen β_0

Chr.lm <- lm(lV ~ lD + lH, data = cherry); Chr.lm0 <- lm(lV ~ offset(2*lD + lH), data = cherry); anova(Chr.lm0, Chr.lm)

Vergleich normales Modell mit Modellidee. P-Wert signifikant? Wenn ja, dann haben wir statistische Evidenz gegen Nullhypothese von oben. Wenn nicht signifikant, dann keine Evidenz gegen Nullhypothese.

Vertrauensintervall

summary(Spr.lm) →

- Vertrauensintervall umfasst alle **Parameterwerte $\boldsymbol{\alpha, \beta}$**, die auf Grund bestimmten statistischen Tests nicht abgelehnt werden.
- Damit Null-Hypothese auf Niveau 5 % nicht abgelehnt werden muss, muss Absolutbetrag Testgrösse kleiner als das entsprechende Quantil sein: $\boldsymbol{|T| \le q_{0.975}^{t_{n-2}}}$, $\boldsymbol{\hat\beta \pm q \cdot se\langle\hat\beta\rangle}$, $\boldsymbol{se\langle\hat\beta\rangle = \frac{\hat\sigma}{\sqrt{SS_X}}}$

Coefficients:

	Schätzer	Standardfehler	Test einzel H_0: $\alpha \vee \beta = 0$		
	Estimate	Std. Error	t value	Pr(>|t|)	P-Wert
(Intercept)	$\hat\alpha$ = 3.89	$se(\hat\alpha)$ = 0.3156	12.36	8.61e-08	
lDist	$\hat\beta$ = -1.92	$se(\hat\beta)$ = 0.1783	-10.79	3.45e-07	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Schätzung der Standardabweichung der Fehler = σ ↓

Residual standard error: 0.1145 on 11 degrees of freedom

Multiple R-squared: 0.913, R^2 Adjusted R-squared: 0.9058

F-statistic: 116.4 on 1 and 11 DF, p-value: 3.448e-07

F-Test einf. Regression H_0 «β gleich 0» gleicher Wert ↑ wie oben

- **confint(Spr.lm, level = 0.95)** **«Hand»:↓** $qt\left(1 - \frac{\alpha}{2}, df = n - 2\right)$

$\alpha = 5\%$	*2.5 %*	*97.5 %*
(Intercept)	*3.204*	*4.594*
lDist	*-2.315*	*-1.531*

coef(fit)[2] + c(-1,1)*qt(*0.975*, *fit*$df.residual)*summary(*fit*)$coefficients[*2*,2] Kof= $\beta_j + 1$

-1.92 + c(-1,1) * 2.200 * 0.1783 = KI (Daten v. summary output oben)

KI von lDist [-2.315, -1.531], womit Steigung -2 gut mit Daten verträglich.

Ziel: Aufgrund Daten etwas über Werte der Parameter des Modells zu sagen, die plausibel erscheinen. Drei Fragen stellen:

1. **Welcher Wert** ist für den (resp. jeden) Parameter am **plausibelsten**? Antwort wird durch **Schätzung** gegeben
2. Ist ein **bestimmter Wert plausibel**? Die **Entscheidung** trifft man **mit** einem **Test**.
3. Welche **Werte** sind **insgesamt plausibel**? Als Antwort erhält man ein **Vertrauens-/Konfidenzintervall.**

Erwartungswert $\mathbb{E}\langle Y\langle x_0\rangle\rangle$ für Wert x_0 (**Punktschätzung/Prognose**): **predict(Spr.lm, newdata = data.frame(lDist = 1))**

Korrektur Prognosewerte von log. Var: (h.p <- predict(Cher.lm, newdata = data.frame(lD = log(5.3), lH = log(27))))
exp(h.p); # → ist median anstelle $\mathbb{E}\langle X\rangle$ versus; **exp(h.p + summary(Cher.lm)$sigma^2/2)); #** ist nun $\mathbb{E}\langle X\rangle$
Korrektur bei log Transformation, gilt für Erwartungswert und Vertrauensintervall, aber nicht für Prognoseintervall (dieses ist eher zu lang als zu kurz). siehe auch Abschnitt «**Auswirkungen der Transformation von Zielvariablen»** in dieser ZF.

Vertrauensintervall VI für η_0

- Oft liegt Interesse an Funktionswert $h\langle x_0\rangle$ an bestimmten Stelle x_0. Bildung Vertrauensintervall.
- $h\langle x_0\rangle = \alpha + \beta x_0 = \eta_0$ (eta). Finden aller plausiblen Werte via eines Tests $\eta_0 = \eta_0^\circ$ $\boldsymbol{se\langle\hat\eta_0\rangle = \hat\sigma\sqrt{\frac{1}{n} + \frac{(x_0 - \bar x)^2}{SS_X}}}$
- Schätzung von $\boldsymbol{\eta_0}$: $\boldsymbol{\hat\eta_0 = \hat\alpha + \hat\beta x_0}$, t-Verteilung mit n-2 Freiheitsgraden Testgrösse $\boldsymbol{T = \frac{\hat\eta_0 - \eta_0^\circ}{se\langle\hat\eta_0\rangle}}$
- Vertrauensintervall für $\boldsymbol{\eta_0 = h\langle x_0\rangle}$: $\boldsymbol{(\hat\alpha + \hat\beta x_0) \pm c_w \cdot se\langle\hat\eta_0\rangle}$, wobei c_w das Quantil der t-Verteilung mit n-2 Freiheitsgraden
- **predict(Spr.lm, newdata = data.frame(lDist = c(log(50), log(100))), interval = "confidence", level = 0.95 , *parm*=2)**
- Kann Intervalle für mehrere Schätzer (parm = weglassen) gleichzeitig erstellt werden. Variablename in Data.frame (hier **lDist**) muss mit Name in Datensatz respektive Befehl **Spr.lm <- lm(lErsch ~ lDist, data = SprengS1)** übereinstimmen.

Prognoseintervall

E_i: $\hat\alpha + \hat\beta x_0 \pm c_w \cdot \hat\sigma\sqrt{1 + \frac{1}{n} + \frac{(x_0 - \bar x)^2}{SS_X}} = \hat\eta_0 \pm c_w \cdot \sqrt{\hat\sigma^2 + (se^2\langle\hat\eta_0\rangle)}$

- In welchem **Bereich** kommt eine **zukünftige Beobachtung** zu liegen. **Aussage** über **Zufallsvariable**! Das **Prognoseintervall berücksichtigt Variabilität** von E_i ↑
- **Beispiel Sprengungen:** Wie gross wird die Erschütterung sein, wenn die Distanz zur Sprengstelle 50m/100m beträgt?
  (h <- predict(Spr.lm, newdata = data.frame(lDist = c(log(50), log(100))), interval = "prediction", level = 0.95))

	fit	lwr	upr
1	1.45428	0.8496141	2.058947

anschliessend **Rücktransformation**, da Werte logarithmiert **exp(h)**

- Tabelle für Modell 95%-Prognose-Intervalle für alle Beobachtungen: **cat.p <- predict(cat.lm, interval = "prediction")**
- **cbind(unten = cat.p["lwr"], fit = cat.p["fit"], oben = cat.p["upr"], laenge = cat.p["upr"] - cat.p["lwr"])**

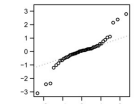
Plot: Forbes.lm <- lm(y ~ x, Forbes[-12,]); x0 <- data.frame(x = seq(min(Forbes$x), max(Forbes$x), length = 50))
Forbes.cia <- predict(Forbes.lm, newdata = x0, interval = "confidence", level = 0.99); plot(Forbes$x, Forbes$y)
lines(x0$x, Forbes.cia["upr"], col = 2); lines(x0$x, Forbes.cia["lwr"], col = 2); abline(Forbes.lm, col = "blue", lty=1)
Forbes.pia <- predict(Forbes.lm, newdata = x0, interval = "prediction", level = 0.99) #Darstellung in Plot: Erww. E(X)
lines(x0$x, Forbes.pia["upr"], col = 7); lines(x0$x, Forbes.pia["lwr"], col = 7) #Vertrauens- und Prognoseband

Prüfen der Modelleignung

- Beim **Regressionsmodell beruhen** die eingeführten Schätz- und Testmethoden auf **Modellannahmen**: Die Fehler E_i sind **unabhängig** und **normalverteilt** mit **konstanter Varianz**, E_i unabhängig $\sim \boldsymbol{\mathcal{N}\langle 0,\ \sigma^2\rangle}$. Annahmen:
  1. **Der Erwartungswert der $\boldsymbol{E_i}$ ist** $\boldsymbol{\mathbb{E}\langle E_i\rangle = 0}$,
  2. **die $\boldsymbol{E_i}$ haben alle die gleiche Varianz** $\boldsymbol{var\langle E_i\rangle = \sigma^2}$,
  3. **sie sind normalverteilt.**
  4. **sie sind unabhängig.**
- Für die Regressionsfunktion wurde Linearität vorausgesetzt. Wenn Annahme verletzt, gilt für Fehler Annahme (1) nicht.
- Damit ist klar, dass alle Verletzungen der gemachten Annahmen in den Fehlern E_i sichtbar sein müssen.
- Gerade aus Regressionsanpassung in Diagramm legen. Eine Möglichkeit, diese Güte der Anpassung zu quantifizieren, ist das Bestimmtheitsmass R^2, das im R-Output mit «**Multiple R-squared**» bezeichnet ist.

Bestimmtheitsmass $\boldsymbol{R^2}$: misst Güte der Anpassung mit Anteil der durch die Regression erklärten Streuung der Y-Werte:

- $\boldsymbol{R^2 = \frac{\sum_{i=1}^n (\hat y_i - \bar y)^2}{\sum_{i=1}^n (y_i - \bar y)^2} = \frac{SS_{Fit}}{SS_Y}}$, R^2 liegen zwischen 0 und 1; je grösser desto besser.
- In Technik und Naturwissenschaften: Werte grösser 0.9 durchaus üblich. In Geistes- und Sozialwissenschaften: um 0.6
- Falls kein Achsenabschnitt β_0 im Modell vorhanden, sollte diese Definition des Bestimmtheitsmasses nicht verwenden.
- **Interpretation**: das Bestimmtheitsmass ist identisch zur **quadrierten Korrelation** zwischen der Zielvariablen Y_i und den angepassten Werten (fitted values): $\boldsymbol{\hat y_i = \hat\beta_0 + \hat\beta_1 x_i}$. Demzufolge miss das Bestimmtheitsmass wie die Korrelation auch die Stärke des linearen Zusammenhangs zwischen der Zielvariablen und der Anpassung.
- **Achtung! Das Bestimmtheitsmass ist kein Mass für die Eignung des Regressionsmodells!**



Langschwänzigkeit →

1.) Tukey-Anscombe-Diagramm (T-A-Diagramm/Plot): $\mathbb{E}\langle E_i\rangle = 0$ → **Erwartungswert = 0**

- Tragen Residuen R_i gegen die angepassten Werte $\hat{y}_i (= \hat{\alpha} + \hat{\beta} x_i)$ auf.
- $\mathbb{E}\langle E_i\rangle = 0$ → **Residuen** sollten im Tukey-Anscombe-Diagramm in allen Abschnitten um **horizontale Null-Linie streuen**.
- **Idee:** Struktur mit dem **Glätter** «loess» mit lokalen Gerade (anstelle Mittelwert) durch **Bootstrap-Simulationen** unter der Gültigkeit der Modellannahme erzeugen. **Vergleich simulierten Kurven** mit **ursprünglichen**, ob «**extremer**»?
- **Vorgehen Bootstrap-Simulationen:** Erzeuge zufällig neue Beobachtungen y_i^*, die dem Modell entsprechen, d.h. erzeuge n $\mathcal{N}\langle 0,1\rangle$-verteilte Zufallszahlen e_i^* und bilde daraus neue Zielvariablenwerte $y_i^* = \hat{y}_i + \hat{\sigma} \cdot e_i^*$.
- Danach Regressionsrechnung mit erklärenden Variablen durchführen und den neu erzeugten Werten y_i^* durch, berechnet den Glätter für das T-A-Diagramm und ihn ins Diagramm einzeichnen. Wiederhole dies beiden Schritte $n = 19$ Mal.
- **Überprüfung in Plot, ob Glättung in Bandbreite der Simulationen liegt und ob Kurvenform untypisch (Banane)!**

2.) Scale-Location Plot / Streuungs-Diagramm: $var\langle E_i\rangle = \sigma^2 = konst$ → **konstante gleiche Varianz**

- Erfassen Struktur mit einem «gleitenden Streumass». Eine einfache Möglichkeit besteht darin, die zuvor benutzte Methode «loess» auf die **Absolutwerte** $|R_i|$ der **Residuen** oder besser auf $\sqrt{|R_i|}$ anzuwenden → **scale-location plot**
- $\sqrt{|R_i|}$ Wurzel verwenden, weil sonst $|R_i|$ zu schief verteilt ist. Der Glätter schätzt σ nicht direkt.

3.) Histogramm/Normal QQ-Plot: $E_i = \mathcal{N}\langle 0, \hat{\sigma}\rangle$ **Fehler sind normalverteilt**

- Besser als Histogramme ist **QQ-Plot** → Quantile der empirischen Verteilung der Residuen werden mit Quantilen der Normalverteilung verglichen. Falls Daten normalverteilt sind, **streuen Punkte** um eine **Gerade**.
- **Achtung:** Ein normal QQ-plot für die Y_i ist sinnlos, da die Y_i ja verschiedene Erwartungswerte haben, weshalb geschätze E_i genommen werden, sprich die Residuen. **| Plot: streuen (nicht) innerhalb stochastischen Fluktuation?**

source("…/RFn_Plot-lmSim.R"); par(mfrow = c(2, 3)); plot(*fit*, *which* = *1:3*); plot.lmSim(*fit* , SEED = 1798, rob = T)

Aussage zu Plots: Der Glätter liegt ((teilweise) nicht) innerhalb der stochastischen Fluktuation der Simulation / die Punkte streuen gut um eine Gerade und liegen (nicht) innerhalb der stochastischen Fluktuation des QQ-Normalplot. Damit liegt (k)eine statistische Evidenz gegen die Annahme Erwartungswert $\mathbb{E}\langle E_i\rangle = 0$ / konstante Varianz $var\langle E_i\rangle = \sigma^2$ / Normalverteilung $E_i \sim \mathcal{N}$ vor. Frage, ob Voraussetzung der Unabhängigkeit erfüllt ist, ist schwieriger zu beantworten

Prüfen der Modelleignung: wollen mit drei Darstellungen sicherstellen, dass Daten keine für Theorie gefährlichen Abweichungen zu Voraussetzungen gibt. Da Residuen Realisierungen von Zufallsgrössen, Voraussetzungen nie «exakt erfüllt».

Skalierte Residuen:

- Wollen die Verteilung der Zufallsfehler E_i überprüfen, haben aber die Residuen R_i benützt – und das ist nicht dasselbe!
- Falls Fehler E_i normalverteilt → Residuen ebenfalls. Aber sie haben **nicht** die **gleiche Varianz**!
- Damit Residuen gleiche Verteilung haben, muss man sie skalieren: $\tilde{R}_i := R_i \Big/ \sqrt{1 - \left(\frac{1}{n} + \frac{(x_i - \bar{x})^2}{SS_X}\right)} \sim \mathcal{N}\langle 0, \sigma^2\rangle$
- **Vorgehen:** Zur Überprüfung der Voraussetzungen verwenden der standardisierten Residuen, - ausser im Tukey-Anscombe-Diagramm, weil so Abweichungen von einem konstanten Erwartungswert besser erkannt werden können.

Behandlung von Unzulänglichkeiten: J.W. Tukey nannte sie First Aid Transformations

- Annahme «Streuung der Zufallsfehler ist konstant» verletzt. **Häufig:** Abhängigkeit der Streuung von $\hat{y}$.
- Die **Streuung nimmt** im Streuungs-Diagramm **zu** und der **Normal-Plot** zeigt eine **rechts schiefe Verteilung** an.
- Folgende Transformationen können helfen:
  - **Logarithmus**-Transformation für **Konzentrationen/Beträge**,
  - **Wurzeltransformation** für **Zähldaten**
  - **Arcus-Sinus-Wurzel-Transformation** $\hat{y} = arcsin\langle\sqrt{y}\rangle$ oder
  - **Logit-Transformation** für **Anteile (Prozentzahlen/100):** $\hat{y} = log\left\langle\frac{y+0.005}{1.01-y}\right\rangle$ **First-Aid/Log Faktor 10**
- Immer anwenden, ausser triftige Gründe sprechen dagegen. **Nicht anwenden bei Zeitvariablen / zu wenig Streuung**

Auswirkungen der Transformation von Zielvariablen: (bei Log Problem für Erwartungswert und Vertrauensintervall)

- Transformation der Zielvariablen ändert Form der Verteilung der Fehler. Beispiel Logarithmus-Transformation:
- Sowohl erklärende Variable als auch Zielgrösse seien Konzentrationen, $\tilde{Y} = log\langle Y\rangle$ und $\tilde{x} = log\langle x\rangle$
- Aus $\tilde{Y}_i = \alpha + \beta\tilde{x}_i + E_i$ wird $log\langle Y_i\rangle = \alpha + \beta \cdot log\langle x_i\rangle + E_i$ oder $Y_i = e^{\alpha} \cdot x_i^{\beta} \cdot e^{E_i}$ **Korrektur** $exp\left\langle\frac{\hat{\sigma}^2}{2}\right\rangle$, für $\hat{y}_0$ → $exp\left\langle\hat{y}_0 + \frac{\hat{\sigma}^2}{2}\right\rangle$
- D.h. haben **Potenzgesetz** für die ursprünglichen Grössen und **Fehler** ist **proportional** (und **nicht additiv**).
- Falls $\beta = 1$ ist die Zielvariable proportional zu x bis auf einen **multiplikativen zufälligen Fehler**. → **Log-Norm-verteilt**
- Bsp: $\log(B_i) = \beta_0 + \beta_1 \cdot \log(N_i) + \beta_2 \cdot \log(C_i) + \beta_3 \cdot p_i + E_i \Leftrightarrow B_i = e^{\beta_0} \cdot N_i^{\beta_1} \cdot C_i^{\beta_2} \cdot e^{\beta_3 \cdot p_i} \cdot e^{E_i}$ $E_i \sim \mathcal{N}(0, \sigma^2)$ unabhängig

Annahme «Erwartungswert ist konstant null» verletzt:

- **Systematische Abweichung** im **Erwartungswert** kann oft durch **Transformation** der **erklärenden Variablen** x **oder**
- durch **Hinzufügen** eines **zusätzlichen Terms** x^2 (**quadratische Regression**) zum **Verschwinden** gebracht werden.

Annahme «Fehler sind normalverteilt» durch Ausreiser verletzt: *weglassen von Beobachtungen mit Bf subset* ↓↓

- Richtigkeit Daten überprüfen! Falls Daten i.O., Transformation der Ziel- und/oder der erklärenden Variablen abklären.

Annahme «Fehler sind normalverteilt» durch Langschwänzigkeit verletzt: lm(y ~ x, Forbes, subset = c(-11, -12))

- Extremste Beobachtungen weglassen, bis Langschwänzigkeit verschwindet → Resultate mit Vorsicht geniessen, weil zu optimistisch! Kleinste-Quadrate-Methode ist bei Langschwänzigkeit nicht optimal, nur **robuste Methoden geeignet**.

Unabhängigkeit der zufälligen Fehler: Korrelation in der Interpretation

- Wenn Beobachtungen eine zeitliche Reihenfolge einhalten, dann könnten **Autokorrelationen** vorhanden sein.
  - Also Residuen R_i in dieser Reihenfolge auftragen – Keine Strukturen sichtbar sein! **plot(resid(fit), type = "h")**
  - R_t gegen R_{t-1} in Streudiagramm auftragen – Punkte sollten frei streuen und **keine Korrelation** zeigen.
  - **fit <- lm(y ~ x, data = dat); x.n < − length(resid(fit)); scatter.smooth(resid(fit)[1:(x.n - 1)], resid(fit)[2:x.n])**
- **Wenn (Auto-) Korrelationen** vorliegen, dann sind **P-Werte** der üblichen Tests häufig grob **falsch** und **Vertrauensintervalle zu kurz**. Methoden, die Korrelationen berücksichtigen: **verallgemeinerte Kleinste Quadrate** (R: gls aus nlme).

Multiple lineare Regression

- **Zusammenhang** zwischen **einer Zielgrösse** und **mehreren erklärenden Variablen** $x^{(1)}, x^{(2)}, \dots, x^{(m)}$. → **pairs(dat)**
- Erweiterung des einfachen Regressionsmodell: $Y_i = \beta_0 + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)} + \dots + \beta_m x_i^{(m)} + E_i$ mit **unab.** $E_i \sim \mathcal{N}\langle 0, \sigma^2\rangle$
- Die (**unbekannten**) **Parameter** sind die **Koeffizienten** $\beta_0, \beta_1, \dots, \beta_m$ zu den **erklärenden Variablen** und die **Varianz** σ^2 der **zufälligen Abweichungen** E_i. Die β sind die **Steigungen** in **Richtung** der **x-Achsen**. Achsenabschnitt: $\beta_0 = \alpha$
- Schätzung Koeffizienten β_j erfolgt via **KQ** und sind **normalverteilt** → Verteilung von Teststatistiken → VI
- Auch die **Streuung** wird auf analoge Weise wie vorher geschätzt $\hat{\sigma}^2 = \frac{1}{n-p}\sum_{i=1}^{n} R_i^2$ $\quad p = Anzahl\ Koeffizienten\ \beta$

Beispiel Sprengungen: Erschütterung = Distanz und Ladung: $\log(ersch)_i = \beta_0 + \beta_1 \log(dist)_i + \beta_2 \log(ladung)_i + E_i$

- **Spr2.lm2 <- lm(lErsch ~ lDist + lLadung, data = SprengS2); summary(Spr2.lm)**
- Zeilen im Summary-Output: Prüft, ob Nullhypothese $\beta_j = \beta$ mit den Daten verträglich ist (die anderen β's sind beliebig!)
- Im Beispiel: Beide Variable haben auf dem 5 % Niveau einen nachweisbaren Effekt auf die Zielgrösse, weil die jeweiligen P-Werte kleiner als das Niveau 5 % sind.
- Die Frage, wie stark $x^{(2)}$ für sich allein, ohne die Konkurrenz von $x^{(1)}$, die Variabilität von Y teilweise erfasst oder nichts damit zu tun hat, lässt sich mit einer **einfachen Regression** beantworten und wird mit multiplen Regressionsrechnung nicht geprüft.
- Multiple Regression ist **nicht Summe einfachen Regressionen**!

```
     Min        1Q    Median        3Q       Max
-1.00326  -0.26665   0.00522   0.2237

Coefficients:
             Estimate  Std. Error  t value  Pr(>|t|)
(Intercept)     6.52      0.5132    12.707   <2e-16 ***
lDist          -1.51      0.1111   -13.592   <2e-16 ***
lLadung         0.80      0.3042     2.658   0.0109 *
---
                                                .1 ' ' 1
Residual standard error: 0.3521 on 45 degrees of freedom
Multiple R-squared: 0.8048    Adjusted R-squared: 0.7962
F-statistic: 92.79 on 2 and 45 DF, p-value: < 2.2e-16
```

Test auf $H_0: \beta_j = \beta = 0$, $\beta_k, k \neq j$ beliebige «optimale» Werte annehmen können

Schätzer $\hat{\beta}_j$ | Sd Fehler ^ $\hat{\beta}_j$ | P-Wert | $\hat{\beta}_0 =$, $\hat{\beta}_1 =$, $\hat{\beta}_2 =$ | $se\langle\hat{\beta}_0\rangle =$, $se\langle\hat{\beta}_1\rangle =$, $se\langle\hat{\beta}_2\rangle =$ | R^2

Schätzung der Standardabweichung der Fehler = σ ↓

F-Test auf «Alle β_j ausser β_0 sind gleich 0.» ↑

- **Nur** eine **multiple Regression zeigt** den **wahren Einfluss** der **einzelnen erklärenden Variablen**.

Vertrauensintervall für β_j - confint(SP.lm2, level = 0.95) #wie bei einfacher Regression

- Verteilung eines einzelnen Koeffizienten β_j: $\hat{\beta}_j \sim \mathcal{N}\langle\beta_j, \sigma^2 V_j\rangle$, wobei Grösse nur von erklärenden Variablen V_j abhängt.
- Also lautet das $(1-\alpha) \cdot 100$ %-Vertrauensintervall: $\hat{\beta}_j \pm c_w \cdot se\langle\hat{\beta}_j\rangle$ mit
  - $se\langle\hat{\beta}_j\rangle = \hat{\sigma}\sqrt{V_j}$ «Std. Error»
  - und c_w dem $(1-\alpha/2)$-Quantil der t-Verteilung mit $(n-p)$ Freiheitsgraden.

Bestimmtheitsmass – «Multiple R-Squared» $\quad R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = \frac{SS_{Fit}}{SS_Y}$

- misst Anteil durch die Regressionsfunktion erklärten Streuung an der Streuung der Y-Werte.
- Ist wie in einfachen Regression definiert, R^2 liegen zwischen 0 und 1; je grösser, desto besser.
- Es entspricht dem Quadrat der Korrelation zwischen den beobachteten y_i und den angepassten Werten $\hat{y}_i$ d.h. es misst den **linearen Zusammenhang**. Die nach **Kleinsten Quadraten** geschätzten Koeffizienten **minimieren** nicht nur die **Quadratsumme** der **Residuen**, sondern **maximieren** auch die **Korrelation** zwischen den **angepassten Werten** und den **Beobachtungen** der **Zielgrösse**. Der **maximale Wert** ist die **multiple Korrelation**.

Vielfalt der Modellierungsmöglichkeiten

- Die erklärenden Variablen müssen von **keinem bestimmten Datentyp sein**, **nicht** einer **bestimmten Verteilung** folgen (sind keine Zufallsvariablen) und es gibt **keine Voraussetzung** über ihre **Abhängigkeit** untereinander.

Polynomiale Regression

- Die x-Variablen können im Modell in irgendeiner Weise aus ursprünglichen erklärenden Variablen abgeleitet werden.
- Z.B. quadratische Regression: $Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + E_i$ mit $x^{(1)} := x$ und $x^{(2)} := x^2$ ergibt sich
- $Y_i = \beta_0 + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)} + E_i$ ein multiples lineares Regressionsmodell.
- In gleicher Weise können auch höhere Potenzen eingeführt werden → polynomiale Regression
- Polynomiale Regression ist ein multiples lineares Regressionsmodell. In der polynomialen Regression wird eine **nicht-lineare Funktion** in x angepasst, die jedoch **linear** in den **unbekannten Parametern** $\beta_0, \beta_1, \beta_2$ ist!

- **Modell:** $\log(P_i) = \beta_0 + \beta_1 \cdot \log(C_i) + \beta_2 \cdot \log(C2_i) + E_i \Leftrightarrow P_i = e^{\beta_0} \cdot C_i^{(\beta_1 + \beta_2 \cdot \log(C2_i))} \cdot e^{E_i}$ $E_i \sim \mathcal{N}(0, \sigma^2)$ unabhängig
- **C$lCarat <- log(C$Carat); C$lCarat2 <- log(C$Carat)^2; C.lm <- lm(lPrice ~ lCarat + lCarat2, data = C)**
- **Linien in Plot: x <- seq(min(C$lCarat), max(C$lCarat), length = 50);** → Bei qu. Regr. min-max-seq von **x** term
- **predict(C.lm, newdata = data.frame(lCarat = x, lCarat2 = x^2), interval = "c")** #nicht von x² bilden und dann erst ^2!

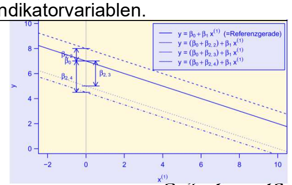
Erklärende Variable ist binär **Spreng2$Stelle <- as.factor(Spreng2$Stelle)**

- Einfache lineare Regression mit einer binärer erklärenden Variablen x: $Y_i = \beta_0 + \beta_1 x_i + E_i$, $i = 1,2,\dots,n$ seien die 15 ersten Elemente in x gleich 0 und die restlichen gleich 1, dann $Y_i = \beta_0 + \beta_1 \cdot \mathbf{0} + E_i$, $i = 1,2,\dots,15$ und $Y_i = \beta_0 + \beta_1 \cdot \mathbf{1} + E_i$, $i = 16,17,\dots,n$. Das Regressionsmodell beschreibt zwei unabhängige Stichproben.
- Der Koeffizient β_1 beschreibt eine zusätzliche Effekt-Verschiebung in der zweiten Stichprobe.

Beispiel Sprengungen: Kann sein, dass örtlichen Gegebenheiten der Messstellen Einfluss auf Erschütterung haben?

- Die Variable, die die Stellen bezeichnet, nennt man **Faktorvariable**. Um sie in ein Regressionsmodell einzubeziehen, führt man für jede mögliche Stelle eine «**Indikatorvariable**» (Dummy Variable) ein: $x_i^{(j)} = \begin{cases} 1 \\ 0 & sonst \end{cases}$ 1 falls i-te Beobachtung aus der j-ten Stelle (Gruppe) → Faktorvariable führt zu einem Block von Indikatorvariablen.

Skizze zur Illustration einer möglichen Parametrisierung von parallelen Geraden



- Betrachten Fall mit Distanz und Stelle. Modell: $Y_i = \beta_0 + \beta_1 x_i^{(1)} + \beta_1 x_i^{(2)} + E_i$, wobei $Y = \log(\text{Erschütterung})$ und $x^{(1)} = \log(\text{Distanz})$ und $x^{(2)}$ die Messstelle bezeichnet
- Sei Stelle 1 die **Referenzstelle** mit $Y_i = \beta_0 + \beta_1 x_i^{(1)} + E_i$. Stellen 2, 3 und 4 sind geschiftete Geraden, die mit Indikatorvariablen $St2, St3, St4$ beschrieben werden.
- Der Einfluss der Stelle 1 ist im Achsenabschnitt β_0 enthalten.

- Das Regressionsmodell mit parallelen Geraden wird geschrieben: $Y_i = \beta_0 + \beta_1 x_i^{(1)} + \beta_{2,2} St2_i + \beta_{2,3} St3_i + \beta_{2,4} St4_i + E_i$.
  $\beta_{2,2}, \beta_{2,3}, \beta_{2,4}$ sind die Verschiebungen der Achsenabschnitten.
- **Spr.lm <− lm(lErsch ~ lDist + lLadung + Stelle, data=Spreng2)**

Coefficients: | Schätzer $\hat{\beta}_j$ | Sd Fehler ^ $\hat{\beta}_j$ | Test auf $H_0: \beta_j = \beta = 0$

	Estimate	Std. Error	t value	Pr(>|t|) P-Wert	
(Intercept)	$\hat{\beta}_0$ = 5.78	$se(\hat{\beta}_0)$ = 0.64967	8.898	3.25e-11	***
lDist	$\hat{\beta}_1$ = -1.33	$se(\hat{\beta}_1)$ = 0.14073	-9.506	4.97e-12	***
lLadung	$\hat{\beta}_2$ = 0.69	$se(\hat{\beta}_2)$ = 0.29666	2.332	0.0246	*
StelleSt2	$\hat{\beta}_{3,2}$ = 0.37	$se(\hat{\beta}_{3,2})$ = 0.17257	2.192	0.0340	*
StelleSt3	$\hat{\beta}_{3,3}$ = 0.04	$se(\hat{\beta}_{3,3})$ = 0.14657	0.341	0.7349	
StelleSt4	$\hat{\beta}_{3,4}$ = 0.25	$se(\hat{\beta}_{3,4})$ = 0.17216	1.482	0.1459	

Schätzung der Standardabweichung der Fehler = σ ↓

Residual standard error: 0.3379 on 42 degrees of freedom

Multiple R-squared: 0.83 = R^2 Adjusted R-squared: 0.8122

F-statistic: 41.66 on 5 and 42 DF, p-value: 3.194e-15

Globaler F-Test auf H_0 «Alle β_j ausser β_0 sind gleich 0.» ↑

Sind zwei Geraden gleich?

- Unterscheiden sich zwei Geraden im Achsenabschnitt/Steigung oder in beidem? Formulieren als Modell:
  $\boldsymbol{Y_i = \alpha + \beta x_i + \Delta\alpha g_i + \Delta\beta x_i g_i + E_i}$, g_i = Gruppenzugehörigkeit
- $g_i = 0$ falls Gruppe A und $g_i = 1$ falls Gruppe B
  Gleichung fallweise aufgeschrieben: $\boldsymbol{g_i = 0: Y_i = \alpha + \beta x_i + E_i}$
  und $\boldsymbol{g_i = 1: Y_i = (\alpha + \Delta\alpha) + (\beta + \Delta\beta) x_i + E_i}$ → Der Term «$x_i g_i$» wird auch als **Wechselwirkung** bezeichnet.
- Die beiden Geraden stimmen in Steigung überein, wenn $\Delta\boldsymbol{\beta} = \boldsymbol{0}$ oder gesamthaft überein, wenn zugleich $\Delta\boldsymbol{\alpha} = \Delta\boldsymbol{\beta} = \boldsymbol{0}$
- Wechselwirkung kann in R mit **lm(Y ~ x1 + x2 + x1 : x2, data = dat)** oder **lm(Y ~ x1 * x2, data = dat)** erfasst werden.

Vergleich von Regressionsmodellen mit F-Test

- Messstellen Einfluss auf Erschütterung? Kein Einfluss → alle **Koeffizienten** Indikatorvariablen **null:** $\Delta_1 = \Delta_2 = \Delta_3 = \Delta_4 = 0$
- $H_0: \beta_{j1} = 0, \beta_{j2} = 0, \dots, \beta_{jq} = 0$, Alternative H_A: mindestens ein $\beta_{jk}, k = 1, \dots, q$, ist ungleich 0. Teststatistik $F := \frac{(SS_E^* - SS_E)/q}{SS_E/(n-p)}$
- $SS_E^* := \sum_{i=1}^n R_i^2$, r_i aus «kleinen» Modell, $(p-q)$ Koeffizienten | $SS_E := \sum_{i=1}^n R_i^2$, r_i aus «grossen» Modell, p Koeffizienten.
- Verteilung von F unter der Nullhypothese $F \sim \mathcal{F}_{q,n-p}$, F-Verteilung mit q und $n-p$ Freiheitsgraden.
- F-Test mit **anova(fit1, fit2)** (Modell mit zusätzlicher erkl. Variable vs. Modell ohne diese) durchgeführt. «ANOVA» = Analysis of Variance, Varianzenvergleich. Alternative: **drop1(fit, test = "F")** → zeilenweiser Test: H_0 «β_j ist gleich 0»

Globaler F-Test – Summary-Output

- F-Test im Summary-Output dreht sich um Frage: Beeinflusst die Gesamtheit der erklärenden Variablen die Zielgrösse?
- **Nullhypothese: $\boldsymbol{H_0}$: Alle $\boldsymbol{\beta_j}$ ausser $\boldsymbol{\beta_0}$ sind gleich 0. Alternative: Mindestens eines dieser $\boldsymbol{\beta_j}$ ist ungleich 0**.
- Testgr.: $F := \frac{(SS_Y - SS_E)/m}{SS_E/(n-p)}$, $SS_Y := \sum_{i=1}^n (Y_i - \bar{Y})^2$, $SS_E := \sum_{i=1}^n R_i^2$, Unter H_0 Testgrösse F -verteilt, $m = p-1$ und $n-p$ FG

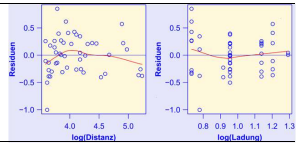
Theoretische Verteilung der Residuen

- Residuen Kleinste-Quadrate-Schätzung sind **normalverteilt,** haben Erwartungswert 0, aber **Varianz** ist **nicht konstant**.
- Damit Residuen gleiche Verteilung haben wie Fehler → skalieren: $\tilde{R}_i := \frac{R_i}{\sqrt{1-H_{ii}}} \sim \mathcal{N}\langle 0,\ \sigma^2 \rangle, i = 1, \dots, n$
- **Stand. Residuen**: $\frac{R_i}{\hat{\sigma}\sqrt{1-H_{ii}}}, i = 1, \dots, n$ (sind weder standardnormalverteil noch t-verteilt!). Residuen sind korreliert.
- In der Residuenanalyse verwenden wir für Überprüfung der Verteilung die standardisierten Residuen, ausser für das T-A-Diagramm, damit man dort die Abweichungen von einem konstanten Erwartungswert besser erkennen kann.

PRESS-Residuen / Studentisierte Residuen

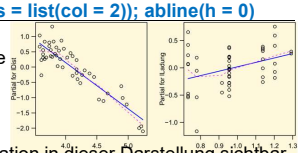
- Residuen, die Vorhersagefehler schätzen (prediction errors). Mit PRESS-Residuen, $\hat{e}_i$, wird Differenz bezeichnet zwischen Y_i und angepassten Wert. Ergibt sich, wenn die i-te Beobachtung zum Anpassen des Modells **nicht** verwendet.
- Residuen muss nicht für jede Beobachtung neu angepasst werden: $\boldsymbol{\hat{e}_i := \frac{R_i}{1-H_{ii}}, i = 1,2,..n}$, Varianz: $\boldsymbol{var\langle \hat{e}_i \rangle = \frac{\sigma^2}{1-H_{ii}}}$
- Standardisierung PRESS-Residuum, Ersetzung σ^2 durch $\hat{\sigma}^2 : \frac{\hat{e}_i}{\hat{\sigma}\sqrt{1-H_{ii}}} = \frac{R_i}{\hat{\sigma}_{(i)}\sqrt{1-H_{ii}}}, \hat{\sigma}^2_{(i)} = \frac{1}{(n-p-2)}\left((n-p-1)\hat{\sigma}^2 - \frac{r_i^2}{(1-H_{ii})}\right)$
- Oder, standardisierte/studentisiertes Residuum: $R_i^* := \frac{R_i}{\hat{\sigma}_{(i)}\sqrt{1-H_{ii}}}$, ist t-verteil mit $(n-p-1)$ Freiheitsgraden

Residuen-Analyse bei der multiplen Regression – **fit <- lm(lErsch ~ lDist + lLadung, data = Spr)**



- Gleiche Analyse mit bekannten drei Diagrammen. Gibt zusätzliche Diagramme.
- Fazit nach Analyse: Welches entscheidendste Unstimmigkeit und wie bereinigen?
- Falls sich im Tukey-Anscombe-Diagramm Abweichungen von der angenommenen Form der Regressionsfunktion zeigen, welche erklärenden Variablen sind zu transformieren? → darum Residuen gegen alle erklärenden Variablen auftragen ↗
- **Plot erstellen: par(mfrow = c(1, 2)); scatter.smooth(dat$lDist, resid(fit) , lpars = list(col = 2)); abline(h = 0)**
- **Plot pro erkl. Variable erstellen scatter.smooth(dat$lLadung, resid(fit) , lpars = list(col = 2)); abline(h = 0)**

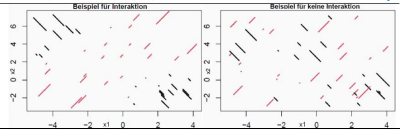
Partial Residual Plots (PRP)



- Andere erkl. Var. haben Einfluss und können Bild (Grafik violett) verfälschen. Abhilfe PRP. Effekt der anderen erkl. Var. wird herausgerechnet: $r_i^{(-k)} = r_i + \hat{\beta}_k x^{(k)}$
- Falls lin. **Regressionsmodell geeignet**, streuen Punkte im Diagramm $r_i^{(-k)}$ gegen
- $x^{(k)}$ um **Gerade** mit **Steigung** β_k. Beispiel für lErsch ~ lLadung + lDist →
- Falls notwendig, die erklärende Variable $x^{(k)}$ zu transformieren, so ist die Transformation in dieser Darstellung sichtbar.

par(…); termplot(fit, partial.resid = T, smooth = panel.smooth, ylim = "free", col.res = 1, col.term = 4, col.smth = 2)

Additivität der erklärenden Variablen



- Die Effekte von zwei erkl. Variablen addieren sich. Streudiagramm:
- $(x_i^{(1)}, x_i^{(2)})$ mit Residuum als strichförmiges Symbol. **Länge** Strich **proportional** zum **Absolutbetrag**, **Steigung** Strich (± 1) fürs **Vorzeichen**
  Interaktion mit zusätzlicher Variable modellieren: $x_i^{(3)} := x_i^{(1)} \cdot x_i^{(2)}$

- **library("sfsmisc"); bsp.lm <- lm(y ~ x1 + x2, dat = bsp); p.res.2x(x = ~ x1 + x2, data = bsp.lm, scol = 2:1)**
- **Alternative: p.res.2x(x = bsp$x1, y = bsp$x2, z = resid(bsp.lm), size = 1.5, slwd = 2)** #size/slwd → klarere Striche

Gewichtete lineare Regression

- Was soll man tun, wenn die **Residuenanalyse** die **Notwendigkeit zeigen** oder die **Art** der **Datenerhebung impliziert**, dass die **Varianzen** der **einzelnen Zufallsfehler** als **nicht konstant** anzunehmen sind? $\sigma_i^2 = var\langle E_i \rangle, i = 1,2, \dots, n$
- **Ansatz:** $\sigma_i^2 = \sigma^2 \cdot \frac{1}{w_i}$, wobei w_i bekannt und σ^2 aus Daten geschätzt werden. Geht auch als Matrix: $\boldsymbol{W} := \begin{bmatrix} \frac{1}{w_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{w_n} \end{bmatrix}$

Beispiel Fettgehalt in Fischen

- Untersuchen, ob sich mittlere Fettgehalt zwischen vier Fischarten unterscheidet. Hierfür werden 3 Fische pro Art zufällig ausgewählt. Von diesen sollen je vier Fleischproben entnommen und mittlere Fettgehalt bestimmt.
- **Problem:** Wegen eines Missgeschicks konnten nicht immer jeweils alle 4 Proben ausgewertet werden.
- **Offensichtlich:** die Proben von einem Fisch sind ähnlicher als die Proben zwischen zwei Fischen. D.h. die Messfehler sind nicht identisch verteilt. **Ausweg:** Analysiere nur den mittleren Fettgehalt pro Fisch. $Fat.mean := \frac{1}{No_i} \sum_{j=1}^{NoS_i} y_{ij}$, wobei y_{ij} der Fettgehalt des j-ten Fisches aus der Art i ist und NoS_i die Anzahl noch Fische der Art i.
- **Konsequenz:** Da der beobachtete mittlere Fettgehalt (Fat.mean) einer Fischart i (Species) auf einer unterschiedlichen Anzahl Proben NoS_i (NoS im Datensatz) basiert, haben die Beobachtungen unterschiedliche Varianzen.
- Die **Varianz** des Mittelwerts ist: $var\langle \bar{y}_i \rangle = \frac{1}{NoS_i} var\langle E_i \rangle = \frac{1}{NoS_i}\sigma^2$, weil die einzelnen Beobachtungen unabhängig sind.
- **Gewichte festlegen:** Mittelwert streut weniger, umso grösser Anzahl Proben NoS_i ist. Deshalb Gewicht $w_i = NoS_i$
- Daraus folgt: $2 \cdot Q\langle \underline{\beta} \rangle = \sum_i w_i R_i^2 = \underline{R}^T \boldsymbol{W} \underline{R}$ bezüglich $\underline{\beta}$ zu minimieren. Schätzung: $\hat{\underline{\beta}} = (\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{W} \underline{Y}$
  entspricht erwartungstreue Regression von $\sqrt{w_i} y_i$ auf $\sqrt{w_i} x_i^{(1)}, \dots, \sqrt{w_i} x_i^{(m)}$, für **Varianz** gilt $var\langle \hat{\underline{\beta}} \rangle = \sigma^2 (\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X})^{-1}$
- Residuen sind: $\underline{R} = (\boldsymbol{I} - \boldsymbol{H_W W}) \underline{Y}$, mit $\boldsymbol{H_W} = \boldsymbol{X}(\boldsymbol{X}^T \boldsymbol{W} \boldsymbol{X})^{-1} \boldsymbol{X}^T$,
- Kovarianzmatrix wird zu $var\langle \underline{R} \rangle = \sigma^2 (\boldsymbol{W}^{-1} - \boldsymbol{H_W})$, auf Diagonalen der Matrix stehen die Varianzen der Residuen.

FFat2.lm <− lm(Fat.mean ~ Species, data = FFat2.Agg, weights = NoS); summary(FFat2.lm) #NoS = Var. in df

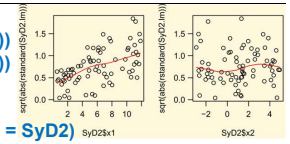
Welche Residuen soll man in grafischen Darstellungen verwenden? Generell gilt: $\tilde{R}_i := \frac{R_i}{\sqrt{\frac{1}{w_i} - (\boldsymbol{H_W})_{ii}}}$

- für Beurteilung der Verteilung und Streuung der Fehler verwendet man **skalierte Residuen**: z.B. im Normalplot, scale-location plot oder im Diagramm Residuen gegen Hebelarme.
- Geht es um Eignung der Regressionsfunktion, werden Darstellungen $\sqrt{w_i} r_i$ gegen $\sqrt{w_i} \hat{y}_i$ oder gegen $\sqrt{w_i} x_i^{(k)}$ betrachtet. In diesen Skalen erfolgt eigentliche Anpassung. Z.B. im T-A-Plot oder im Diagramm Residuen gegen erkl. Variablen.

Weitere Motivation für Gewichte: Residuen-Analyse zeigt **Varianzen** der Zufallsfehler $\sigma_i^2 = var\langle E_i \rangle$ **nicht konstant ist**.

- Dann ist allerdings die Herausforderung, die **relativen Genauigkeiten** v_i für jede Beobachtung i zu finden, sodass $var\langle E_i \rangle = \sigma^2 v_i$ bestimmt werden kann.
- **Falls** sich in einem **Streudiagramm** der **Residuen** gegen ein $X^{(j)}$ zeigt, dass die **Streuung** von $X^{(j)}$ **abhängt**,
  - dann kann man versuchen, eine **Funktion** $v\langle X^{(j)} \rangle$ **anzugeben**, die diese **Abhängigkeit beschreibt**.
  - dann **Anwendung** der **gewichteten Regression** mit **Gewichten:** $w_i = \frac{1}{v\langle x_i^{(j)} \rangle}$
  - σ^2 ist dann die **Varianz jener Beobachtung**, bei der $v_i = 1$ ist.

Beispiel: Hängt die Varianz von einer erklärenden Variablen ab?



- **scatter.smooth(SyD2$x1, sqrt(abs(rstandard(SyD2.lm))), lpars = list(col = "red"))**
- **scatter.smooth(SyD2$x2, sqrt(abs(rstandard(SyD2.lm))), lpars = list(col = "red"))**
- Ja, die Streuung hängt von der erkl. Var. x1 ab.
- d.h. mit steigenden x1-Werten nimmt die Streuung proportional zu

Probieren deshalb Ansatz: SyD2.lm2<- lm(y ~ x1 + x2, weight = 1/SyD2$x1^2, data = SyD2)

- Gewichtete Regression führt zu besserem Modell als ohne Gewichte. (Kontrolle mit den drei bekannten Plots)

Beispiel: Produktion mit drei Geschwindigkeiten: 100, 150, 200. Wie viele Defekte gibt es? #Var nicht konstant in Scale-P
Modell: g.lm <- lm(DEFECTS ~ SPEED, data = G); (G.var <- aggregate(resid(g.lm) ~ G$SPEED, FUN = var));** #Ausprobieren was hilft (Werte ca. gleich) **→ G.var[,2]/c(100,150,200)^2; G.var[,2]/(c(100,150,200)/100)^2** #passt am besten **(G$w <- 1/((G$SPEED/100)^2)); g.lmw <- lm(DEFECTS ~ SPEED, data = G, weights = w) #Modell passt nun

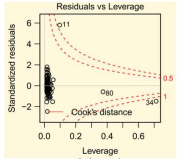
Einflussreiche Beobachtungen

- **Thema Ausreisser:** Die Antwort auf die Frage, ob eine Beobachtung ein Ausreisser sei, hängt vom Modell ab!
- Wie stark beeinflusst eine Beobachtung (Ausreisser) Analyse? → Analyse ohne die fragliche Beobachtung wiederholen und Effekte auf Schätzung, Tests, Vertrauensintervalle, ..., untersuchen
- Effekt messen mit **Cooks Distanz**: misst Veränderung aller angepassten Werte $\hat{y}_i$ beim Weglassen i-ten Beobachtung.
- $d_i = \frac{1}{p \cdot \hat{\sigma}^2} \cdot \sum_{j=1}^n \left(\hat{y}_j^{(-i)} - \hat{y}_j\right)^2, i = 1, \dots, n$, wobei $\hat{y}^{(-i)}$ angepasster Wert bei Weglassen i-ten Beobachtung ist $d_i = \frac{1}{p} \tilde{R}_i^2 \frac{H_{ii}}{1-H_{ii}}$, mit $\tilde{R}_i = \frac{R_i}{\hat{\sigma}\sqrt{1-H_{ii}}}$
- **Faustregel: Zu einflussreiche Beobachtung = Cook's Distanz grösser als 1**

Hebelarm H_{ii}: Diese Grösse misst, wie «untypisch» die Beobachtung in Bezug auf die erklärenden Variablen ist.

- Wertebereich von H_{ii}: $0 \le H_{ii} \le 1$, Varianz der Residuen: $var\langle R_i \rangle = (1 - H_{ii})\sigma^2$ → **zu einflussreiche Beobachtung?**
- Wenn ein Wert Y_i um Δy_i verändert, dann misst $H_{ii}\Delta y_i$ die Veränderung des zugehörigen angepassten Wertes $\hat{y}_i$.
- **Faustregel I**: Beobachtungen mit einem Hebelarm $H_{ii} > 2\frac{p}{n}$ haben zu grosse Hebelwirkung **plot(fit, which = 4)**
- **Faustregel von Huber:** Beob. mit Hebelarm unter 0.2 sind unbedenklich; jene mit über 0.5 sollten vermieden werden
  Achtung: Hebelpunkte H_{ii} und Cook's Distanz untersuchen Effekt **nur einer** Beobachtung!

Beispiel: Es sind zwei **zu einflussreiche** (Cook's Distanz > 1) **Beobachtungen** sichtbar:



- Beobachtung 11 ist wegen **grobem Ausreisser** (in y-Richtung) zu einflussreich,
- Beobachtung 34 ist wegen zu **grossen Hebelarm** ($\boldsymbol{h_i > 0.2}$) zu einflussreich,
- Beobachtung 80, ist ein weiterer, jedoch ungefährlicher Hebelpunkt.

Robuste Anpassungsmethoden

- In **normaler** Residuen Analyse werden **eventuelle Ausreiser** und **grosse Hebelpunkte gefunden** und **entfernt**.
- **Robuste Methode:** findet **Ausreiser leichter** und können diese aufgrund der Robustheit der Schätzer **stehen lassen**.
- **Ausreisser** stellen nur eine «**kleine**» **Abweichung** von den Annahmen dar, aber mit **verheerender Wirkung**.
- Oft existiert keine vernünftige Alternative zur Normalverteilung oder zur Linearität. Deshalb geht man davon aus, dass
  - das **Regressionsmodell** für die **Mehrheit** der **Beobachtungen stimmt**,
  - es aber **möglich** ist, dass in **wenigen Fällen Abweichungen** davon **auftreten können**.
- Gesucht ist Schätzverfahren, die Parameter so schätzt, wie wenn Abweichungen nicht vorhanden sind → robust
- Erster informeller Ansatz zur Robustheit ist, **erstens** die **Daten** auf **offensichtliche Ausreisser** zu prüfen, **zweitens** diese zu **löschen** und **drittens** das zum **Modell** optimale **Schätzverfahren** auf die bereinigten Daten anzuwenden.
- Dieses Verfahren ist allemal besser, als die Ausreisser zu ignorieren. Die Resultate sind jedoch immer zu optimistisch.

- Ob Schätzer **robust** ist, kann mit **zwei Massen** untersuchen. Beide Masse beruhen auf Idee, **Verhalten** einer **Schätzfunktion** unter dem **Einfluss** von **groben Fehlern** (gross errors), d.h. willkürlichen hinzugefügten Daten, zu studieren:
  - **Einflussfunktion** (influence function): Die Sensitivität (gross error sensitivity) basiert auf der Einflussfunktion und misst den maximalen Effekt einer einzelnen Beobachtung auf den Schätzwert.
  - **Bruchpunkt** (breakdown point): Der Bruchpunkt gibt den **minimalen Bruchteil** von **Beobachtungen** an, der genügt, um **unglaubwürdige Schätzwerte** zu erhalten. Bruchpunkt misst **Grösse** der **Störung**, die zur **Katastrophe** führt.
- **Schätzer guten Robustheitseigenschaften:** beschränkte Sensitivität und Bruchpunkt möglichst nahe an max. Wert $\frac{1}{2}$.

Konstruktion eines robusten Regressionsschätzer

- **Idee:** Gewicht der Ausreiser mittel gewichteter linearer Regression herunterzugewichten respektive beschränken.
- Gewichtete Normalgleichung $\boldsymbol{X}^T\boldsymbol{W}\underline{R} = \underline{0}$ respektive $\sum_{i=1}^{n} w_i \cdot R_i \cdot \underline{x_i}$, mit $R_i = Y_i - \underline{x_i}^T\underline{\beta}$ zeigt, dass Beobachtungen i mit grossen Residuen R_i mit entsprechenden Gewichten w_i heruntergewichtet werden.
- Geeignete Gewichte mittels Einflussfunktion IF bestimmen $IF\langle y\rangle = w_i \cdot r_i \cdot \boldsymbol{M} \cdot \underline{x_i}$, wobei $\boldsymbol{M}$ noch zu bestimmende Matrix.
- Wollen Einfluss grosser Residuen beschränken, muss $\psi\langle r_i\rangle := w_i \cdot r_i$ beschränkt werden. Gewicht mit $w_i = \frac{\psi\langle r_i\rangle}{r_i}$ bestimmen. Da Gewicht von Residuen abhängt und damit vom wahren Parameter $\underline{\beta}$ mit $r_i = y_i - \underline{x_i} \cdot \underline{\beta}$, können wir Gewicht nicht mehr explizit berechnen. Das führt dazu, dass Lösung nur mit iterativen Algorithmus gefunden kann und Standard-Fehler der Schätzungen anders berechnet werden müssen als in der gewichteten linearen Regression.

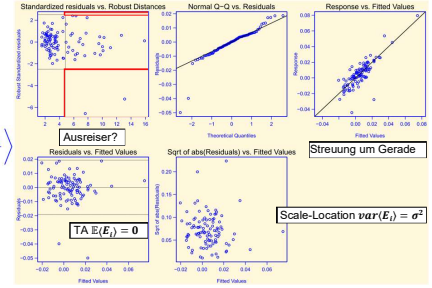
Robuster Regressions-M-Schätzer (ist Schätzer mit oben beschriebenen impliziten Gewichten)

- In Funktion ψ gibt es einen Knick, welcher festlegt, von welcher Stelle extreme Beobachtungen an Einfluss verlieren.
- Sinnvoll wenn Fehler in ihrer Einheit $\frac{r_i}{\sigma}$ gemessen werden. Knickstelle c gegeben durch theoret. Effizienz $c = 1.345$
- M-Schätzgleichung: $\sum_{i=1}^{n} \psi\langle \tilde{r}\langle \hat{\underline{\beta}}_M\rangle\rangle \cdot \underline{x_i} = \underline{0}$, mit $\tilde{r}_i = \frac{Y_i - \underline{x_i}^T\hat{\underline{\beta}}_M}{\sigma}$ oder mit Gewichten: $\sum_{i=1}^{n} w_i \cdot r_i \cdot \underline{x_i} = \underline{0}$, mit $w_i = \frac{\psi\langle \tilde{r}\langle \hat{\underline{\beta}}_M\rangle\rangle}{\tilde{r}_i}$
- Die Standardabweichung als Schätzung des Skalenparameters (Streuung) σ ist jedoch extrem unrobust. Eine bessere robustere Schätzung für den Skalenparameter ist zum Beispiel der **median of absolute values**: $\hat{\sigma}_{MAV} := \frac{med_i\langle|r_i|\rangle}{0.6745}$
- Wenn sich ein Ausreisser auf Hebelpunkt befindet, erweist sich die Regressions-M-Schätzung als **ungenügend robust**.
- Dieser Schätzer begrenzt nur Einfluss Residuen, nicht Einflussfunktion als Ganzes, weshalb sie Bruchpunkt von 0 hat.

library("MASS"); f.rlm <- rlm(Y ~ X1 + X2, method = "M", data = dat); summary(f.rlm); par(mfrow = c(2, 2)); plot(f.rlm)

Regressions-MM-Schätzung (Regressions-Modified-M-Estimator)

- Um Schätzer zu konstruieren, der robust mit hohen Bruchpunkt ist, müssen Beobachtungen mit grossen Residuen nicht nur heruntergewichtet, sondern allmählich (in Abhängigkeit des Betrages der Residuen) ignoriert werden.
- Da ψ-Funktion wieder auf die Nulllinie absinkt, werden solche Regressions-M-Schätzern als redescending bezeichnet.
- Da Schätzfunktion (viele) lokale Minima besitzt, ist Optimierung schwierig. Lösung Problem: mit etwas anderen Optimierungsaufgabe mit stochastischem Suchalgorithmus werden gute Startwerte für Koeffizienten $\underline{\beta}$ gefunden.
- Die Lösung ist als S-Schätzer (ist Startwert) bekannt, hat jedoch schlechte theoretische Eigenschaften.
- Die zum Startwert nächst gelegene lokale Lösung des eigentlichen Problems führt dann zu einem besseren Schätzer, wobei als Schätzwert für σ jene aus den Startwerden, d.h. $\hat{\sigma}_0$ verwendet wird.
- Lösung hat hohen Bruchpunkt von $\epsilon^* = \frac{1}{2}$ und die Inferenz des Schätzers kann analog zu M-Schätzern gemacht werden.
- Chance, aus allfälligen Abweichungen besseres Modell entwickeln. Da robuste Anpassungsmethoden durch «Verunreinigungen» der Daten kaum irritiert werden, sind Modellabweichungen viel leichter in der Residuen-Analyse zu erkennen.
- Schätzer approximativ normalverteilt: $\left(\hat{\beta}_M\right)_j \sim \mathcal{N}\langle\beta_j, \sigma^2\tau V_j\rangle$, mit $\tau > 1$ Korrekturfaktor. Approximation besser, je grösser n
- Vertrauensintervall $\left(\hat{\beta}_M\right)_j \pm c_w \cdot se\langle\left(\hat{\beta}_M\right)_j\rangle$ mit $se\langle\left(\hat{\beta}_M\right)_j\rangle = \hat{\sigma}_0\sqrt{\hat{\tau}V_j}$, c_w Quantil t-Verteilung mit $n - p$ Freiheitsgraden und $\hat{\sigma}_0$ der Residuenstandardfehler vom initialen S-Schätzer ist.
- Robuste Version des F-Tests ist eine $\Delta^* \sim \chi_q^2$ Verteilung unter der H_0 $D\left\langle \underline{y}, \hat{\underline{\theta}}_{MM}\right\rangle := 2 \cdot \hat{\sigma}_o^2 \cdot \sum_{i=1}^{n} \rho\left\langle \frac{y_i - \underline{x_i}^T\hat{\underline{\theta}}_{MM}}{\hat{\sigma}_o}\right\rangle$



library("robustbase"); #KS2014 für effizientes Vertrauensintervall
f.lmrob <- lmrob(Y ~ X1 + X2, setting = "KS2014", data = dat)
par(mfrow = c(2,3)); plot(f.lmrob, las = 1)

➔ Zweck Robuste Plots: Identifizierung der Ausreiser
➔ sehen hier im Plot überall zwei Ausreiser

Variablenselektion und Modellbildung

- Welche erklärenden Variablen sollen wie ins Regressionsmodell eingehen? Warum kein Modell mit allen erkl. Var.?
  - **Einfachheit** – gibt es mehrere Erklärungen, so sollte man die Einfachste wählen.
  - **Reduktion Schätzvariabilität** – unnötige erklärende Variablen verschlechtern Genauigkeit der Modellschätzung.
  - **Bessere Interpretierbarkeit** – Multikollinearität führt zu nicht eindeutigen Lösungen.
  - **Vorhersage** – weniger erklärende Variablen, weniger Aufwand beim Vor- und Aufbereiten der Daten.
- Ist ein bestimmter Term $\beta_j x^{(j)}$ im Modell nötig? Nützlich? Überflüssig? → Grundbaustein für die Variablenselektion
- Als Hypothesen-Prüfung diese Frage schon gelöst: Nullhypothese $\beta_j = 0$ prüft (t-Test) → multiples Testproblem
- Weiteres Problem: müsste Voraussetzungen über Fehler prüfen, wenn P-Werte der Tests zum Nennwert nehmen wollte.
- **Lösung:** Statt Tests für strikte statistische Schlüsse zu verwenden, nehmen P-Werte der t-Tests.
- Bei Faktorvariablen Ersetzung t-Test durch F-Test: **add1(..., scope=..., test = "F")** oder **drop1(..., test = "F")**

Manuelle Vorwärts-Selektion (Forward Selection)

Mit diesem auf P-Werten gestützten Kriterium können wir eines der Variablenselektionsverfahren wie folgt formulieren:

- Wir starten mit dem einfachsten kleinen Modell: $Y_i = \beta_0 + E_i$ **AKW.lm0 <- lm(Y ~ 1, ...)**
- Ersten Schritt wird die Variable in das Modell aufgenommen, die in einfacher Regression den **kleinsten P-Wert** besitzt.
- Dann wird jeweils die nächste noch nicht berücksichtigte Variable mit dem kleinsten P-Wert aufgenommen.
- **Stoppregel:** Grundsätzlich nicht festlegbar. Schranke 0.05 pro Zeile ist üblich, problematisch da multiples Testproblem

AKW.lm0 <- lm(lgK ~ 1, data = AKW) #Beginn mit **Zielvariable ~ 1**, einfaches Modell erstellen
add1(AKW.lm0, scope = ~ lgG + D + WZ + BZ + Z + NE + KT + BW + sqrtN + KG, test = "F") #add1 Fun wiederholen
AKW.lm0 <- update(AKW.lm0, .~. + KG); drop1(AKW.lm0, test = "F") #Hinzufügen Variable + Überprüfung p-Wert
add1(… genau wie oben); AKW.lm0 <- update(AKW.lm0, .~. + lgG); drop1(AKW.lm0, test = "F") # p-Werte i.O.?
…add1(… genau wie oben); AKW.lm0 <- update(AKW.lm0, .~. + KT) drop1(AKW.lm0, test = "F") # p-Werte i.O.?
AKW.lm0 <- update(AKW.lm0, .~. -KT) #KT Variable nun nach drop1() nicht mehr signifikant, deshalb wieder entfernen
summary(AKW.lm0) → gibt nun das Modell an, AKW.lm0 ist nun folgendes Modell: lgK ~ KG + lgG + D + NE

Manuelle Rückwärts-Selektion (Backward Selection)

Zweiter Vorschlag für ein Variablenselektionsverfahren, das auf den P-Werten gestützt ist, lässt sich wie folgt formulieren

- Wir starten mit dem «vollen» Modell aller erklärenden Variablen $Y_i = \sum_{k=0}^{m} \beta_k x_i^{(k)} + E_i$
- In den weiteren Schritten wird jeweils **eine Variable aus** dem **Modell genommen**. Es ist jeweils die erklärende Variable, die am Variablenselektion mit P-Werten «unwichtigsten» ist und damit den **grössten P-Wert** hat.
- **Stoppregel:** Grundsätzlich wieder nicht festlegbar. Oft sieht man die Schranke 0.05 (**P-Wert für jede Zeile**)

AKW.lm0 <- lm(lgK ~ lgG + D + NE + KG + BW + BZ + Z + sqrtN + KT, data = AKW) → zuerst volles Modell erstellen
drop1(AKW.lm0, test = "F"); AKW.lm1 <- update(AKW.lm0, . ~ . -BW) #Variable mit höchstem P-Wert entfernen!
…drop1(AKW.lm0, test = "F"); AKW.lm0 <- update(AKW.lm0, . ~ . -KT) #Variable mit höchstem P-Wert entfernen!
drop1(AKW.lm0, test = "F") → nun alle Werte signifikant. AKW.lm0 ist nun folgendes Modell lgK ~ lgG + D + NE + KG

- I.A. **nicht zwingend**, dass beide Selektionsverfahren (Vorwärts- und Rückwärtsselektion) zum **gleichen Modell** führen.
- Achtung, nur eine Variable aufs Mal ausschliessen! Verbleibenden Prädiktoren haben kleinere p-Werte als volles Modell.
- Die Vorhersagekraft konzentriert sich von ursprünglich vielen auf die nun selektierten Variablen.
- Für Vorhersagen ergeben sich oft zu kleine Modelle, die nicht optimale Prognosen geben.
- Entfernte Variablen können kausalen, jedoch kleinen, Einfluss auf die Zielgrösse haben, der im Rauschen untergeht.

Schrittweise Selektion – Kombination aus Vorwärts- und Rückwärtselimination

- Wird in jedem Schritt ausprobiert, ob Weglassen oder Hinzunehmen einer erkl. Var. Modellwahlkriterium verbessert.
- Jene Aktion, die zur grössten Verbesserung im gewählten Modellwahlkriterium führt, wird durchgeführt.
- Verfahren wird beendet, sobald keine Verbesserung mehr möglich ist.

Modellwahlkriterien

- **Variablenselektionsverfahren** beruhen auf **Modellwahlkriterien**, die **Güte** der **Modellanpassung** durch statistische **Masszahl** beschreibt. Die Masszahl sollte **Modellgenauigkeit** und die **Modellkomplexität** bewerten.

In der linearen Regression mit dem KQ-Anpassungskriterium kann

- Modellgenauigkeit mit Summe der **Residuenquadrate** • Modellkomplexität mit **Anz**. Koeffizienten gemessen werden

Ziele Modellkomplexität und **Modellgenauigkeit** sind **konträr**. **Zusätz.** erkl. Var. **führen** zu **höheren Modellgenauigkeit**.

Korrigiertes Bestimmtheitsmass (adjusted R^2_{adj}):

- Reguläres R^2 wird immer grösser je mehr Variablen → ungeeignet da keine Berücksichtigung der Modellkomplexität.
- Deshalb **korrigiertes R^2**: n = Anzahl Beobachtungen, p = Anzahl zu schätzenden Koeffizienten β_k $R^2_{adj} = 1 - \frac{\frac{SS_E}{(n-p)}}{\frac{SS_Y}{(n-1)}} = 1 - \frac{n-1}{n-p}(1 - R^2) = 1 - \frac{\hat{\sigma}^2}{\frac{SS_Y}{(n-1)}}$, wobei $\frac{SS_E}{(n-p)} = \hat{\sigma}^2$

Mallow's C_p-Statistik: minimiert den Vorhersagefehler $C_p := \frac{SS_E}{\hat{\sigma}^2_*} + 2p - n = (n-p)\left(\frac{\hat{\sigma}^2_p}{\hat{\sigma}^2_*} - 1\right) + p$

- $\hat{\sigma}^2_p = \frac{SS_E}{(n-p)}$ = Schätzung $\hat{\sigma}^2$ aus Modell mit p Parametern, $\hat{\sigma}^2_{p^*}$ = Schätzung $\hat{\sigma}^2$ aus grösstem Modell mit p^* Parametern

Das Informations-Kriterium von Akaike (AIC): $AIC = -2\,(max\ LogLikelihood) + 2(Anz.\,geschätzter\ Parameter)$

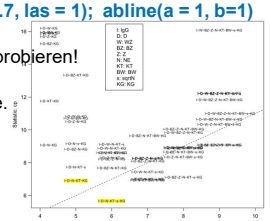
- $AIC = n \cdot log\left(\frac{1}{n}SS_E\right) + 2p^* + Konstante$, p^* = Anz. gesch. Parameter inkl. $\hat{\sigma}$, ◦ = Diamant. AIC ist gut verallgemeinerbar

In R können diese drei Variablenselektionsverfahren **basierend** auf dem **AIC** mit **step(...)** durchgeführt werden:

- **Vorwärts: M0 <- lm(Y~1, data=DF); step(M0, scope=list(lower=~1, upper=~ x1+x2+x3+x4), direction = "forward")**
- **Rückwärts: MF <- lm(Y ~ x1 + x2 + x3 + x4, data = DF); step(MF, *direction = "backward"*)**
- **Schrittweise: Mm <- lm(Y ~ x2, data = DF); step(Mm, scope = list(lower = ~ 1, upper = ~ x1 + x2 + x3 + x4))**
- Letztes Modell in Ausgabe ist jeweils das beste Modell. ↑ scope = range zwisch. kleinsten/grössten Modell

Eindeutige Lösung? Optimales Modell?
- Häufig führen verschiedenen Selektionsarten zu unterschiedlichen Ergebnissen.
- Grund: Keines dieser Variablenselektionsverfahren garantiert, dass gemäss Modellwahlkriterium **global optimale Modell** gefunden wird. → Können auf dem Weg in **lokalen Optimum festhangen** bleiben.
- Einzige Möglichkeit global optimiertes Modell finden, ist alle Modellvarianten zu berechnen. → **All-Subset Selection**
- Bei **Vollsuche** ist der Rechenaufwand zu gross → kombinatorische Explosion, 2^m = Anz. Modelle, m = Anz. erkl. Var.

Vollsuche (All-Subset Selection) mit C_p-Statistik
- **Vorteil:** Falls Modell alle notwend. Var. enthält, dann ist erwartete Kriteriumswert $\mathbb{E}\langle C_p\rangle$ = Anz. Parameter p, $\mathbb{E}\langle C_p\rangle = p$
- Fehlen notwendige Variablen, so ist $\mathbb{E}\langle C_p\rangle \gg p$ **library(leaps); library(car)** #Plot Streudiagramm C_p gegen p
- **AKW.Cp <- regsubsets(lgK ~ lgG + D + WZ + BZ + Z + NE + KT+BW + sqrtN + KG, nbest=6, nvmax=10, data=AKW)**
- **h <- subsets(AKW.Cp, statistic = "cp", legend = "top", min.size = 4, cex.subsets = 0.7, las = 1); abline(a = 1, b=1)**
- Oder par(mai = c(1, 1, 1, 3)) und legend = "interactive", falls Legende Kombi überdeckt
- **nbest**: Anzahl Modell mit p Koeffizienten, die berücksichtigt werden (default = 1) → ausprobieren!
- **nvmax:** Modelle mit maximal p = nvmax werden berücksichtigt (default = 8)
- subsets() zählt Achsenabschnitt **nicht**, deshalb **abline(a = 1, b = 1)** = Winkelhalbierende.
- **statistic** hat noch Argumente "bic" (Bayes Information Criterion) und "adjr2" (R^2_{adj})
- **In Frage** kommenden Modelle müssen um **Winkelhalbierende** $C_p = p$ **streuen** und **min. C_p-Wert und min. p Wert** haben. **#legend = "interactive"** für Plot



Anmerkung zur Variablenselektion
Falls nicht Vollsuche verwendet werden kann:
- Schrittweise Variablenselektion mit möglichst **grossem Modell** als **Start**, da der Suchprozess etwas umfassender ist.
- Vorwärtsselektion vor allem bei Datensätzen mit **vielen erklärenden Variablen** und **wenig Beobachtungen**.

Bei Modellen mit **Faktorvariablen**, **Polynom** oder **Wechselwirkungstermen** sind folgende Regeln zu beachten:
- Bei **Faktorvariablen** werden nicht einzelne Faktorstufen **entfernt**, sondern nur die **ganze Variable**.
- **Wechselwirkungsterme** immer mit **zugehörigen Haupteffekte** im **Modell behalten**.
- Polynomen immer alle Terme bis max. Ordnung im Modell behalten, d.h. Var.selektion bestimmt nur max. Ordnung.
- **Vorsicht** bei **manueller Variablenselektion**. **step(), drop1(), add1()** befolgen diese Regeln**, nicht aber regsubsets().**

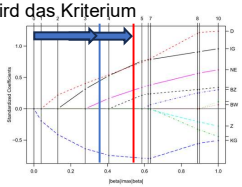
Alternative zu AIC: Bayes Informationskriterium (BIC)
- $BIC = n \cdot log\langle\frac{1}{n}SS_E\rangle + log\langle n\rangle \cdot p^{\diamond} + Konstante$, $p^{\diamond}$ = Anz. gesch. Parameter inkl. $\hat{\sigma}$, $\diamond$ = Diamant.
- **step(..., k = log(nrow(dataset)))** #Rest ist identisch wie bei AIC, Angabe von k führt zu BIC

AIC oder BIC? – Beide führen zu ähnlichen Entscheidungen. BIC bestraft grössere Modelle «härter»
- BIC, wenn verstehen will, welche Prädiktoren Beitrag leisten und wenn schlankes, gut interpretierbares Modell will.
- AIC, wenn Modell für Vorhersage von zukünftigen Werten will, die verwendeten Prädiktoren aber weniger zentral sind.

Welches Modell ist das richtige? Das «beste» Modell ist noch lange nicht das «richtige» oder «wahre» Modell!
- Deshalb immer mehrere Modelle in Betracht ziehen, die Kriterien «gut» – nicht viel schlechter als «beste» – bewerten.
- Wie viel schlechter? – Leider gibt Theorie darauf keine Antwort. Auch berücksichtigen ist **Kollinearität** unter erkl. Var.

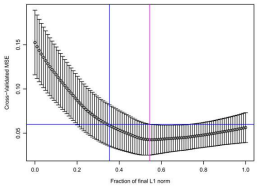
Alternative Ansätze für Variablenselektion
- Heute gibt es sehr viele Daten, auch hochdimensionale Daten und mehr Variablen als Beobachtungen im Datensatz.
- Solche Daten können mit Regressionsmodell analysiert werden, da oft nur wenige erkl. Var. Einfluss auf Zielvar. haben.
- Technik, für solche Situationen, finden sich unter Begriff **«Shrinkage Schätzung»** oder **Regularisierungsmethode**.
- Sie stellt sicher, dass es nicht zu **«Overfitting»** kommt (d.h. das Regressionsmodell wird nicht zu gut an die Daten angepasst, sodass es das Rauschen als systematischen Effekt interpretiert und zu unsicheren Vorhersagen führt)

LASSO-Methode – «Least Absolute Shrinkage and Selection Operators»
- ist eine Vertreterin Shrinkage-Schätzer, die zusätzlich **implizit** noch die Eigenschaft hat, **Variablen** zu **selektionieren.**
- Shrinkage-Schätzer **schrumpft Koeffizienten** in **Richtung** gegen **0**. Wenn Ihr **Beitrag** zur **Erklärung** der **Zielvariable klein** ist, wird beim LASSO der entsprechende Koeffizient auf 0 gesetzt und hat so **Eigenschaft Variablenselektion**.
- Kriterium KQ wird durch «Bestrafungsterm» für Grösse der Koeffizienten ergänzt. Somit wird das Kriterium $Q\langle\underline{\beta};\lambda\rangle := \sum_{i=1}^{n}\tilde{R}_i^2 + \lambda\sum_{k=1}^{m}|\beta_k|$ mit $\tilde{R}_i := \tilde{y}_i - \underline{\tilde{x}_i^T}\,\underline{\beta}$ **minimiert**, Zielvariable ist zentriert: $\tilde{y}_i := y_i - \bar{y}$, Vektor $\underline{\tilde{x}_i}$ aus standardisierten erkl. Var. besteht: $\tilde{x}_i^{(k)} := \frac{x_i^{(k)} - \bar{x}^{(k)}}{sd\langle x_i^{(k)}\rangle}$ und Vektor $\tilde{x}_i$ keine 1 als erstes Element hat (Achsenabschnitt) **par(las = 1, cex.axis = 0.5)**
- **library(lars); h.x <- model.matrix(~ -1 + lgG + D + WZ + BZ + Z + NE + KT + BW + sqrtN + KG, data = AKW); AKW.lasso <- lars(x = h.x, y = AKW$lgK); plot(AKW.lasso); legend("topleft", legend = AKW.lasso$actions)**
  → zwei optimale Parameter $\tilde{b}$ von Plot auf der nächste Seite (gleiche x-Achsenskala, hier ca. 0.35 und 0.55) hier auf Grafik legen. Dann erkl. Variable bis zum Strich von **links** nach rechts geben das Modell.



Optimalen Bestimmung des tuning Parameters $\tilde{b}$ ↘ Modell: (lgK~KG + D + IG + NE) und **(lgK~KG + D + IG + NE + KT)**
- üblicherweise mittels fünf- bis zehnfachen Kreuzvalidierungsverfahren. Das heisst bei fünffachen Kreuzvalidierung,
  - Fünftel der Daten weggelassen und auf restlichen Daten Lasso-Verfahren für ein Raster von $\tilde{b}$-Werten durchführen.
  - Anschliessend wird für jede Lasso-Lösung, indiziert über die verschiedenen $\tilde{b}$-Werte, Prognosen für die ausgelassenen Beobachtungen gemacht und mit den Zielwerten verglichen.
  - Dies wird dann für die anderen 4 Pakete von einem Fünftel der Beobachtungen wiederholt.
- Daraus kann man dann den mittleren Prognosefehler und eine Schätzung dessen Streuung bestimmen.

Das optimale $\tilde{b}$ ist dort, wo
- entweder der **mittlere Prognosefehler minimal** ist (rote Linie in Grafik)
- oder der **mittlere Prognosefehler kleiner** als der **minimale Prognosefehler plus** eine **Standardabweichung** ist. (blaue Linien in Grafik) #Befehle oben gehören auch dazu
- **library(lars); set.seed(4567); AKW.lasso.cv <- cv.lars(x = h.x, y = AKW$lgK, K = 10)**
- **(h.wMin <- which.min(AKW.lasso.cv$cv)); (h.sel <- which(AKW.lasso.cv$cv <= (AKW.lasso.cv$cv[h.wMin] + AKW.lasso.cv$cv.error[h.wMin]))[1])**
- **abline(v = AKW.lasso.cv$index[c(h.wMin, h.sel)], col = c(2, 3))**
- **abline(h = AKW.lasso.cv$cv[h.wMin] + AKW.lasso.cv$cv.error[h.wMin], col = 3)**
- **c(AKW.lasso.cv$index[h.sel], AKW.lasso.cv$cv[h.sel])** #Erster Wert ist das optimale b
- **plot(AKW.lasso, las = 1); abline(v = AKW.lasso.cv$index[c(h.wMin, h.sel)], col = 4, lwd = 5)**
- **Koeffizientenschätzung an Stelle optimalen b: coef(AKW.lasso, s=AKW.lasso.cv$index[h.sel], mode="fraction")**



Kollinearität
- **Hohe Korrelationen** zwischen erkl. Var. oder allgemeinere Formen davon (Kollinearität) sind von **Theorie zugelassen**!
- Führt aber zu Problemen bei Interpretation und beim statistischen Modellieren. Im Vorwärts- und Rückwärts-Verfahren kann es allenfalls vom Zufall abhängen, welche der beteiligten Variablen als erste weggelassen/aufgenommen wird.
- Bei Kollinearität gilt: Eine erkl. Var. lässt sich annähernd als Linearkombination der anderen darstellen, $x_i^{(j)} \approx \gamma_0 + \sum_{k\neq j}\gamma_k x_i^{(k)}$, **gilt** die **Beziehung exakt**, dann gibt es **keine eindeutige Lösung** bei der **KQ-Schätzung**.

Variance inflation factor (VIF)
- Ist die Beziehung $x_i^{(j)} \approx \gamma_0 + \sum_{k\neq j}\gamma_k x_i^{(k)}$ nur annähernd erfüllt, können wir das auch als Regressionsproblem ansehen.
- Das Bestimmtheitsmass der Regression von $x^{(j)}$ auf alle übrigen erkl. Var. (R_j^2 genannt) zeigt,
  - **wie stark** eine solche **Beziehung** ist und
  - ist sinnvolles **Mass** für **Kollinearität**,
  - das erst noch angibt, **welche Variable** «das **Problem** verursacht».
- Ein daraus abgeleitetes Mass für Kollinearität ist der $VIF_j = \frac{1}{1-R_j^2}$
- **Faustregel:** Falls diese Grösse grösser als **5 bis 10** ist, dann sind Probleme mit der Kollinearität vorhanden.
- **library(car); round(vif(AKW.lmV), 2); pairs(AKW);** #Streudiagramm oder **library(ellipse); plotcorr(cor(AKW))**

Weitere Masse zur Identifizierung von Kollinearität
- Die **Konditionszahl (*)** eignet sich für **Entdeckung** von **Multikollinearitäten**. Sie ist definiert als $\mathcal{K} = \frac{\lambda_{max}}{\lambda_{min}}$, wobei λ_{max} der maximale und λ_{min} der minimale Eigenwert von $\boldsymbol{x}^T\boldsymbol{x}$ sind. Da $\boldsymbol{x}^T\boldsymbol{x}$ symmetrisch ist, sind alle Eigenwerte reell.
- Liegt die Konditionszahl **zwischen 100** und **1'000**, so liegt eine **moderate** bis **starke Multikollinearität** vor.
- Ist sie **grösser** als **1'000**, so ist die **Multikollinearität schwerwiegend**.

Auswirkungen der Kollinearität
- **Hohe Kollinearität** führt zu **grossen Standardfehlern** bei den **geschätzten Koeffizienten**. Man kann zeigen, dass die Standardfehler für die jeweiligen Koeffizienten β_j mit $\sqrt{VIF_j}$ aufgeblasen werden. (Optimal wäre ein Faktor von 1.)
- Oft sind deshalb viele/alle Variablen gem. t-Test nicht signifikant. Können jedoch gem. F-Test nicht alle Var. weglassen.
- In gewisse Richtungen sehr **kleine Prognosefehler**, in **andere** sehr **grosse**.
- **Effekte** der einzelnen **Variablen** auf die **Zielvariablen** können **nicht** richtig **interpretiert** werden.
- **Kollinearität muss bereinigt werden**, damit die **Effekte** und ihre statistische Signifikanz **interpretierbar** werden.

Was tun gegen Kollinearität?
- Wenn immer möglich, soll man Beobachtungen so durchführen, dass das Problem vermieden wird. Ansonsten:
  - Variablen linear transformieren; d.h. z.B. stark korrelierte Variablen **ersetzt** man z.B. durch ihre **Summe** und ihre **Differenz**, falls sinnvoll (u.a. gleiche Einheiten), es können auch andere (lineare) Funktionen verwendet werden:
  - **Weitere Möglichkeiten: Mittelwert** (0.5*(…)) oder **Verhältnis** (Relation) bilden.
  - **Jet$mean <- 0.5*(Jet$x1 + Jet$x2); Jet$dif <- Jet$x1 - Jet$x2**
  - **Relation: seatpos$rSeated <- seatpos$Seated / seatpos$Ht; seatpos$rArm <- seatpos$Arm / seatpos$Ht**
  - **Variable** mit dem **höchsten VIF** aus dem Modell **entfernen** (= **«Aputation»**)
- Korrelationen treten fast zwingend auf, wenn **Vergleich** zur **Anzahl Beobachtungen viele erkl. Var**. vorhanden sind.
  → Setze so genannte Shrinkage-Schätzer ein wie z.B.
  - Hauptkomponentenregression (principal component regression), ridge regression oder LASSO, elastic net.
  - Solche Schätzer haben sich vorteilhaft bei Prognosemodellen erwiesen. Allerdings leidet Interpretierbarkeit stark.

Strategien zur Modellbildung

Warum kein Modell mit allen erklärenden Variablen?
- **Einfachheit** – gibt es mehrere Erklärungen, so sollte man das einfachste Modell wählen.
- **Reduktion der Schätzvariabilität** – unnötige erkl. Var. führen zu höheren Variabilität in Schätzung Modellkoeffizienten.
- **Bessere Interpretierbarkeit** – Multikollinearität führt zu nicht eindeutigen Lösungen.
- **Vorhersage** – weniger erkl. Var., weniger Aufwand.

Automatisierte Variablenselektionsverfahren genügen nicht immer, weil:
- Auswahl Variablen ist vom Zufall abhängig → neben «besten» Modell die «fast gleich guten» auch in Betracht ziehen.
- Das «beste» Modell muss nicht unbedingt alle Modellvoraussetzungen erfüllen. → **Residuenanalyse**.
- Können nicht alle nützlichen Transformationen von Beginn weg berücksichtigen wegen grosse Anzahl Möglichkeiten.
- Manchmal liefern die Verfahren Modelle, die mit dem gesicherten Fachwissen nicht übereinstimmen.
- Die «besten» Modelle können für einen Zweck sehr tauglich sein, jedoch für einen anderen unbrauchbar.

Strategien zur Modellbildung: Grobe Skizze zum Vorgehen

i. **Problem verstehen** – Gibt es schon Modellansätze? Effektschätzen oder Vorhersage?
ii. **Daten beschaffen**, kennenlernen und aufbereiten
   - Umgang mit fehlenden Werten
   - Bedeutung der Zahl 0 in den verschiedenen Variablen vereinheitlichen
   - Daten gem. first-aid Transformationen behandeln, ausser es gibt triftige Gründe dagegen; Interaktionen sinnvoll?

iii. **Erste Anpassung** (mit allen Variablen); vorzugsweise mit robusten Methoden oder mit GAM und dann Linearisieren
iv. **Residuen-Analyse**; Tragen Daten dazu bei, das Problem zu lösen? ev. zurück nach ii. oder i.
v. **Variablenselektion,** ggf. Kollinearitäten behandeln
vi. **Modelleignung klären**
   - Residuen-Analyse mit selektionierten Modell
   - Modell/geschätzte Parameterwerte mit Fachwissen abgleichen
   - «out-off-sample» Validierung in Betracht ziehen (Validierung mit noch nicht verwendeten Daten)

Ursache / Kausalität: Grundlegend für alle Wissenschaften ist die Suche nach Ursache-Wirkungs-Beziehungen.

- Aus stat. Korrelationen kann nicht auf solche Beziehungen geschlossen werden!
- Ursachen kann mit Regres.modell von Beobachtungsstudien nicht identifizieren, aber mit Daten geplanten Versuchen.
- Dennoch besteht eine wichtige Anwendung der Regression gerade darin, **Indizien** für solche Beziehungen zu sammeln.
- Die folgenden zwei Arten von Schlüssen sind üblich.
  - Nachweis für vermutete ursächliche Wirkung der Eingangsgrösse auf die Zielgrösse, falls ein Koeffizient in einem Regressionsmodell signifikant von Null verschieden ist und eine ursächliche Wirkung der Zielgrösse auf die Eingangsvariable ausgeschlossen werden kann (Erschütterung kann Distanz zum Sprengort nicht beeinflussen!)
  - Oft Korrelation zwischen Eingangsvariable und Zielgrösse durch dritte Grösse zustande → **Scheinkorrelation**

Additive Modelle

- Möglichkeit, eine geeignete **nicht**-lineare Funktion h anzupassen → «Nichtlinearen Regression»
- Allerdings muss eine solche Funktion vom **Fachwissen motiviert** werden. Wenn dieses Wissen fehlt, was dann?
- Möchten **funktionalen Zusammenhang** h aus Daten bestimmen, ohne spezifische Annahmen an h ausser «**Glattheit**».
- Solche Schätzverfahren für h heissen **Glätter/Nichtparametrische Schätzung** des funktionalen Zusammenhangs.

Polynomiale Regression: Jede stetige Fun. kann **Polynom approximieren**. Je höher Ordnung, je bessere Approximation

- **plot(accel ~ times, data = mcycle); h.new <- data.frame(times = seq(min(mcycle$times), max(mcycle$times), length = 100));** #x-Werte für Approximation **MU.lm3 <- lm(accel ~ poly(times, 3), data = mcycle)** #3 = Polynomgrad
- **lines(h.new$times, predict(MU.lm3, newdata = h.new)); MU.lm6 <- lm(accel ~ poly(times, 6), data = mcycle)**
- **lines(h.new$times, predict(MU.lm6, newdata = h.new), col = 2); legend("topright", legend=c(3,6), text.col=c(1,2))**
- Polynome mit höheren Ordnungen führen zu unpassenden Schwingungen an den beiden Enden → Alternativen gefragt.
- **Zwei Konzepte**, um glatte Funktionen an Daten anzupassen → **Spline-Interpolation/lokalen Regression**.

Splines

- Splines sind **stückweise Polynome** der Ordnung k. Die Verbindungspunkte der Stücke werden als **Knoten** bezeichnet.
- I.A. müssen Funktionswerte und die ersten $(k-1)$ Ableitungen an den Knoten übereinstimmen, damit der Spline eine **kontinuierliche** Funktion mit $(k-1)$ kontinuierlichen Ableitungen ist. Kubische Spline $(k=3)$ ist meistens ausreichend.
- Um Idee in Datenanalyse zu übertragen, hilft es, Spline-Kurve als Linearkombination von q Basisfunktionen dazustellen:
- Legen $(q+3)$ Stützstellen fest, sodass $x_1^* < x_2^* < \cdots < x_{q+3}^*$. Der für Berechnung relevante Bereich liegt in $[x_2^*,\ x_{q+1}^*]$.

B-Spline-Basisfunktionen: Kubische Spline-Kurve: $g\langle x\rangle = \beta_1 \cdot b_1^{(2)}\langle x\rangle + \beta_2 \cdot b_2^{(2)}\langle x\rangle + \cdots + \beta_q \cdot b_q^{(2)}\langle x\rangle$.

- **library(splines); plot(accel ~ times, data = mcycle)** #h.new gleich wie bei Polynom, degree = 3 → Default, kubisch
- **MU5 <- lm(accel ~ bs(times,df = 5, degree=3), data = mcycle); lines(h.new$times, predict(MU5, newdata = h.new))**
- **#Knoten: h.knots <- seq(min(mcycle$times), max(mcycle$times), length=6)[-c(1,6)]; MU.k <- lm(accel ~ bs(times, knots = h.knots), data = mcycle); lines(h.new$times, predict(MU.k, newdata = h.new))** #knots = Innereknoten

Regressions-Splines

Obiger Ansatz in Regression übertragen. Dort rauschen die y_i → besser Daten zu glätten und nicht interpolieren. Deshalb

- Ein Raster von q Stützpunkten wird gewählt.
  - Üblicherweise werden die **Stützstellen** auf **regelmässigen Gitter** über dem Wertebereich von x gewählt oder
  - die Stellen werden durch **Quantile** definiert.
- Glattheit wird über Anzahl Stützpunkte q bestimmt. Ist q klein, ist Kurve sehr glatt, ist q gross, so führt zu Interpolation.

Smoothing Splines – Wie soll q (d.h. df) – und damit die Glattheit der Spline-Kurve – gewählt werden?

- Lösung: Feste, jedoch unterglättende Spline-Approximation. Bei Minimierung der Kriteriumsfunktion Bestrafung für «Welligkeit» (Komplexität) hinzufügt. Anstelle von $\left(\underline{y}-\underline{\underline{x}}\underline{\beta}\right)^T\left(\underline{y}-\underline{\underline{x}}\underline{\beta}\right)$ wird $\left(\underline{y}-\underline{\underline{x}}\underline{\beta}\right)^T\left(\underline{y}-\underline{\underline{x}}\underline{\beta}\right)+\lambda\cdot(Welligkeit)^2$ minimiert.
- Bestrafung $(Welligkeit)^2$ kann als quadrierte Form der Koeffizienten geschrieben werden, $(Welligkeit)^2 = \underline{\beta}^T\underline{\underline{S}}\underline{\beta}$.
- Matrix $\boldsymbol{S}$ wird aus Basisfunktionen abgeleitet. Zielkonflikt Modellgenauigkeit/–glattheit wird durch Parameter λ geregelt.
- Ist λ sehr gross → erhalten wir eine Gerade (glatteste Funktion). Ist $\lambda = 0$ → unterglättete Smoothing-Spline-Kurve.
- Diese beiden Extreme von λ sind nicht praxistauglich. Es gibt ein geeignetes $\lambda \in (0,\infty)$, das prakt. Anforderung genügt.
- Optimale Glattheit mit optimalen Wahl λ, welche **Kreuzvalidierungsverfahren** basiert. ↓ ohne Angabe df ist optimal

- **library(splines); plot(accel ~ times, data = mcycle); MU.ss <- smooth.spline(x = mcycle$times, y = mcycle$accel)**
- **lines(MU.ss); MU.ss5 <- smooth.spline(x = mcycle$times, y = mcycle$accel, df = 5); lines(MU.ss5, col = 2)**

Thin plate regression splines (nachfolgend «tprs» genannt) **library(mgcv); gam(y ~ s(x), data = dat)**

- Vorgestellt Ansatz hat Verbesserungspotential, da man Stützstellen wählen muss → Willkür beim Anpassungsprozess.
- Ansatz mit Basisfunktionen geht nur für **eine** erkl. Var. **Ausweg:** tprs → **knotenfreien Basisfunktionen**, **mehr. erkl. Var.** glätten und ist «**optimal**». **Nachteil:** Schwer darstellen und nicht zu rechenintensiv Implementation haben Approx.

Lokale Regression – LOWESS (LOcally WEighted Scatterplot Smoother)

- LOWESS beruht auf Anwendung der einfachen Regressionsrechnung in einer lokalen Umgebung.
- Idee: kann jede Funktion h in einer kleinen Umgebung von einem vorgegebenen Punkt linear approximieren.
- Wählen ein Fenster um einen Punkt z_1 aus, bei dem wir $h\langle z_1\rangle$ bestimmen wollen.
- Wählen Fenster so, dass die Funktion h möglichst gut durch eine Gerade approximierbar wird.
- Passen Gerade an Punkte an, die im Fenster liegen und machen damit Vorhersage an Stelle z_1 → schätzt Wert $h\langle z_1\rangle$
- Vorgehen wird für Menge von Punkten $z_i,\ i = 1,\dots,N$, die Bereich der x-Werte möglichst gut überdecken, durchgeführt.
- Man erhält dann für die Punkte $z_1,\dots,z_N$ die entsprechenden Vorhersagen $\hat{h}\langle z_1\rangle,\dots,\hat{h}\langle z_N\rangle$.
- Um Funktion h zu visualisieren, werden Punkte durch Geradenstücke verbunden.
- Idee wird mit einer gewichteter KQ-Methode umgesetzt: $\underline{\hat{\beta}}(z_1) = \arg\min_{\underline{\beta}} \sum_{i=1}^n K\left\langle\frac{x_i-z_1}{b}\right\rangle\left(y_i-\left(\beta_0+\beta_1(x_i-z_1)\right)\right)^2$
- Dabei ist b **halbe Fensterbreite** (bandwidth) und $K\left\langle\frac{x_i-z_1}{b}\right\rangle$ **Gewichtsfunktion** (Kerngewicht/kernel weight) für Punkt i.
- **Offene Punkte**: **Wahl Fensterbreite** b, **Wahl Gewichtsfunktion** $K\langle u\rangle$ und **Behandlung Ausreissern** y-Richtung.

Wahl der Fensterbreite b: Wenn wir b sehr klein wählen,

- wird der **Approximationsfehler** sehr **klein** sein – was vorteilhaft ist.
- **Nachteil grosse Varianz** der Vorhersage $\hat{h}\langle z_1\rangle$, weil Anzahl Punkte im Fenster vergleichsweise klein sein wird.

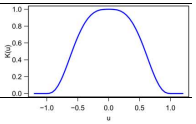
Machen wir die Fensterbreite b **zu gross**,

- dann **können** wir $\hat{h}\langle z_1\rangle$ sehr **genau schätzen**, d.h. mit **kleiner Varianz**,
- jedoch kann dann die **lineare Approximation ungenügend** sein, d.h. der **Approximationsfehler** wird **gross**.

Um sicherzustellen, dass immer genügend Punkte im Fenster sind, machen wir die Fensterbreite folgendermassen:

- Fensterbreite b so wählen, dass vorgegebene Anzahl d Punkte x_i im Fenster liegen. Also ist b für eine Stelle z_1 der d-kleinste Wert unter $|x_i - z_1|$, für $i = 1,\dots,n$.
- Empfohlen wird für die Anzahl d der Wert $\boldsymbol{d = \frac{2}{3}\cdot Anzahl\ Punkte\ n}$. Hilfreich ist es, einige **Werte auszuprobieren**.

Gewichtsfunktion $K\langle u\rangle$



- Die eigentliche Wahl der Gewichtsfunktion K ist weder theoretisch noch empirisch sehr wichtig.
- Kann K so wählen, dass z.B. die Berechnung vereinfacht wird.
- Im LOWESS/LOESS wird der «tricube kernel» benutzt: $K\langle u\rangle = (\max\{1-|u|^3,\ 0\})^3$

Umsetzung mit LOWESS / LOESS

- $\underline{\hat{\beta}}(z_1) = \arg\min_{\underline{\beta}} \sum_{i=1}^n \boldsymbol{w_r\langle x_i\rangle}\cdot K\left\langle\frac{x_i-z_1}{b}\right\rangle\left(y_i-\left(\beta_0+\beta_1(x_i-z_1)\right)\right)^2$ Robustheitsgewichte $w_r\langle x_i\rangle$ sind nur implizit bestimmt.
- Zur Erinnerung: Tukey's bisquare Gewichtsfunktion lautet $w_r\langle x_i\rangle = \left(\max\left\langle 1-\left(\frac{\tilde{r}_i}{c}\right)^2, 0\right\rangle\right)^2$ mit $\tilde{r}_i = \frac{(y_i-\hat{h}(x_i))}{\hat{\sigma}_{MAV}}$
- c kontrolliert Einfluss «schlechte» Beobachtung. Je kleiner c, desto schneller verlieren extreme Beobach. an Einfluss.
- Preis für (zu) kleine c: Effizienzverlust und grosse Standardabw. der Schätzungen → Cleveland schlägt $c = 4.05$ vor.
- **Lokale KQ: scatter.smooth(P$prestige, P$income, span = 0.35, degree = 1, family = "gaussian")**
- **Robust: lines(loess.smooth(P$prestige, P$income, span = 0.35, degree = 1, family = "symmetric"), col = 2)**
- **Alternative: MU.lr <- loess(accel ~ times, data = mcycle); lines(hx$times, predict(MU.lr1g75, newdata = h.new))**
- **Default-Werte:** span = 0.75 → Fensterbreite | degree = 1 → lokal linear, Alternative = 2 → lokal quadratisch | family = "gaussian" → Gauss'sche Abweichung, Alternative "symmetric" = robuste Anpassung/langschwänzige Abweichungen

Das additive Modell

- Bei generalisierte additive Modell (GAM) wird (generalisierte) lineare Regressionsmodell dadurch erweitert, dass **einige** oder **alle linearen Terme** des linearen Prädiktors durch **geeignete glatte Funktionen** der Terme **ersetzt** werden.
- Ersetzung lineare Prädiktor $\eta_i = \beta_0 + \sum_{k=1}^m x_i^{(k)}\beta_k$ durch additiven Prädiktor $\eta_i = \sum_{k=1}^m f_k\langle x_i^{(k)}\rangle$
- **Einfachste**, glatte Funktionen sind **linear**. Führt zu üblichen lin. Modellen. I.A. Funktionen $f_k\langle\cdot\rangle, k = 1,\dots,m$, **unbekannt**.
- Schätzung glatte, unbekannte Funktionen führt zu Streudiagrammglätter → Smoothing-Splines oder LOWESS/LOESS
- Smoothing-Splines wird mit mehreren Variablen mit tprs durchgeführt (R Bf siehe oben → tprs).

- **LOWESS: library(gam); P.gam <- gam(prestige ~ lo(income) + lo(education), data = Prestige); summary(P.gam)**
- Wenn Variablen auf Signifikanzniveau signifikant sind, haben wir einen linearen und/oder nichtlinearen Effekt.
- **lo(income, span = 0.5, degree = 1)** #Default-Werte | span = Glättung, degree = Ordnung verwendete Polynome

```
Anova for Parametric Effects = linearer Teil
               Df  Sum Sq Mean Sq F value    Pr(>F)
lo(income)      1 14899.4 14899.4  304.20 < 2.2e-16 ***
lo(education)   1  8159.2  8159.2  166.58 < 2.2e-16 ***
Residuals      92  4535.5    49.0
---
Anova for Nonparametric Effects = nichtlinearer Teil
            Npar Df Npar F    Pr(F)
(Intercept)
lo(income)      4.0  5.4878 0.0005086  ***
lo(education)   2.4  3.3730 0.0309103  *
```

- **Residuenplot 95%-Vertrauensband: par(mfrow=c(2,2)); plot(P.gam, se = T); plot(P.gam, se = T, residuals = T)**
- **Faustregel:** Falls sich keine Gerade zwischen Vertrauensband legen lässt, so ist **Nichtlinearität statistisch relevant**.
- **Residuen-Plot für GAM-Anpassung: par(mfrow = c(2, 2)); stats:::plot.lm(P.gam)** #gleich wie reg. Residuenanalyse
- **Smoothing-Splines:** es wird kubische Splines mit Knoten an Stellen x_i eingesetzt. Steuerung **Glättungsgrad** über df.
- df = **äquivalente Freiheitsgrade** und ist **Mass** für **Glattheit** der Kurve. df entspricht der Spur der Glättungsmatrix $\boldsymbol{S}$.
- **library(gam); P.gam <- gam(prestige ~ s(income) + s(education), data = Prestige); s(income, df = 4)** #4 = Default
- **Residuenplot 95%-Vertrauensband: par(mfrow=c(2,2)); plot(P.gam, se = T); plot(P.gam, se = T, residuals = T)**

Anmerkungen GAM – Hauptnutzen GAM: Aufdecken von **geeigneten Transformationen** der **erklärenden Variablen**.

- GAM schätzt diese Transformationen (d.h. $f_k(.)$) **nichtparametrisch** (d.h. bestimmt sie Daten gestützt).
- Analyse der notwendigen Transformationen erfolgt über partiellen Residuenplot der geschätzten Funktionen.
- Plot.gam zeigt für jeden additiven Term das Streudiagramm, mit Anwendung Glätter, und die resultierende glatte Kurve.

Stand der Arbeit

SW	Vorlesung	AB/Aufgabe	Skript	ZF
1	☒	☒	☒	☒
2	☒	☒	☒	☒
3	☒	☒	☒	☒
4	☒	☒	☒	☒
5	☒	☒	☒	☒
6	☒	☒	☒	☒
7	☒	☐	☒	☒
8	☒	☐	☒	☒
9	☒	☐	☒	☒
10	☒	☐	☒	☒
11	☒	☐	☒	☒
12	☒	☐	☒	☒
13	☒	☐	☒	☒
14	☒	☐	☒	☒

Woche	Unterricht	Skript	Praktikum
Woche 1 (KW 08)	Einführung und einfache lineare Regression	Kap. 1 bis 2.4	PA 1
Woche 2 (KW 09)	Vertrauens- und Prognosebereiche; Residuen-Analyse	Kap. 2.5 bis 3.1	PA 1, PA 2
Woche 3 (KW 10)	Residuen-Analyse, Behandlung von Unzulänglichkeiten	Kap. 3.1 bis 3.4	PA 2
Woche 4 (KW 11)	Multiple lineare Regression: Die multiple Erweiterung	Kap. 4.1 und 4.2	PA 3
Woche 5 (KW 12)	Multiple lineare Regression: Schätzung und Inferenz	Kap. 4.2 und 4.5	PA 3
Woche 6 (KW 13)	Prüfen der Modelleignung	Kap. 5.1 und 5.2	PA 4
Woche 7 (KW 14)	Gewichtete Regression, einflussreiche Beobachtungen	Kap. 5.3 und 5.4	PA 4
Woche 8 (KW 15)	Robuste Anpassungsmethoden	Kap. 5.5	PA 4
Woche 9 (KW 16)	Variablenselektion, Zwischenprüfung	Kap. 6.1, 6.2/6.A	PA 5
Woche 10 (KW 17)	Variablenselektion (II)	Kap. 6.2 bis 6.3	PA 5
Woche 11 (KW 18)	Kollinearität, Strategien bei der Modellentwicklung	Kap. 6.4 und 6.5	PA 5
Woche 12 (KW 19)	Spline, lokale Regression	Kap. 7.1 bis 7.3	PA 6
Woche 13 (KW 20)	Additive Modelle, Fallstudie (nicht im Skript)	Kap. 7.4	PA 6
Woche 14 (KW 21)	Ausblick, Besprechung Zwischenprüfung, Fragestunde	Kap. 8	PA 6

Modulauflagen

Erlaubte Hilfsmittel für Modulendprüfung: 6 A4-Seiten Zusammenfassung auf Papier (sprich 3 A4 Seiten doppelseitig), einschliesslich R-Befehle. Es sind keine weiteren Hilfsmittel und elektronische Kommunikationsmittel wie z.B. Mobiltelefone und WLAN erlaubt.